Tivoli System Automation for Multiplatforms 4.1.1

Installation and Configuration Guide

SC34-2699-04

Note!

Before using this information and the product it supports, read the information in “Notices” on page 123.

This edition of *System Automation for Multiplatforms Installation and Configuration Guide* applies to Version 4, Release 1, Modification 0 of IBM Tivoli System Automation for Multiplatforms, program number 5724–M00, and to all subsequent releases and modifications of this product until otherwise indicated in new editions.

This edition replaces SC34-2699-01.

IBM® welcomes your comments. A form for readers' comments may be provided at the back of this publication, or you may address your comments to the following address:

IBM Deutschland Research and Development GmbH
Department 3282
Schoenaicher Str. 220
D-71032 Boeblingen
Federal Republic of Germany

FAX (Germany): 07031 16-3456
FAX (Other Countries): 49 7031 16-3456

Internet e-mail: eservdoc@de.ibm.com

If you would like a reply, be sure to include your name, address, telephone number, or FAX number.

Make sure to include the following in your comment or note:

- Title and order number of this book
- Page number or topic related to your comment

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Tables

About this guide

This guide explains how to implement and use the policy-based automated recovery capabilities that are provided by IBM Tivoli System Automation for Multiplatforms (System Automation for Multiplatforms).

System Automation for Multiplatforms provides high-availability for resources on AIX® clusters (on IBM System p), Linux® clusters (on IBM System x, System z®, System i®, and System p), and Windows clusters (on IBM System x).

Who should use this guide

This guide is intended for system administrators and operators who want to use the automation and failover capabilities of System Automation for Multiplatforms.

Where to find more information

The Tivoli System Automation library comprises the following books, including this publication, describing Tivoli System Automation for Multiplatforms:

- *System Automation for Multiplatforms Administrator's and User's Guide*, SC34-2698-01
- *Tivoli System Automation for Multiplatforms Installation and Configuration Guide*, SC34-2699-01
- *Tivoli System Automation for Multiplatforms Reference Guide*, SC34-2700-01
- *Tivoli System Automation for Multiplatforms High Availability Policies Guide, SC34-2660-01*

You can download the complete documentation at

```
http://www.ibm.com/support/knowledgecenter/SSRM2X/welcome
```

The Tivoli System Automation library contains the following books, including this one, describing System Automation Application Manager:

- *System Automation Application Manager Administrator's and User's Guide*, SC34-2701-00
- *System Automation Application Manager Installation and Configuration Guide*, SC34-2702-00
- *System Automation Application Manager Reference and Problem Determination Guide*, SC34-2703-00

You can download the books at:

```
http://www.ibm.com/support/knowledgecenter/SSPQ7D/welcome
```

The IBM Tivoli System Automation home page contains useful up-to-date information, including support links and downloads for maintenance packages. You will find the IBM Tivoli System Automation home page at:

```
www.ibm.com/software/tivoli/products/sys-auto-multi/
```

Conventions

The following highlighting conventions are used in this book:

Table 1. Highlighting conventions used in this book	
Bold	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Also identifies graphical objects such as buttons, labels, and icons that the user selects.
Italic	Identifies parameters whose actual names or values are to be supplied by the user.

Table 1. Highlighting conventions used in this book (continued)	
monospace	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you should actually type.

This manual uses symbols to show resources, resource groups, equivalencies, and relationships. The symbols used are as follows:

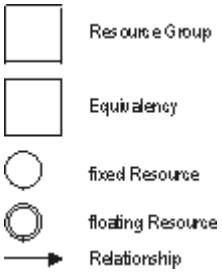

Figure 1. Symbols used in this guide

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Related RSCT information

The following IBM Reliable Scalable Cluster Technology (RSCT) publications are available on the System Automation for Multiplatforms CD:

- *RSCT Administration Guide*
- *RSCT for AIX 5L: Technical Reference*
- *RSCT for Multiplatforms: Technical Reference*
- *RSCT Messages*
- *RSCT Diagnosis Guide*

For more information about RSCT, see IBM Cluster systems.

For more information, see the Linux on IBM zSeries and S/390®: High Availability for z/VM® and Linux IBM Redpaper.

How to obtain publications

The System Automation for Multiplatforms publications are also available (valid at the time of release) at these Web sites:

```
www.ibm.com/servers/eserver/clusters/library/
www.ibm.com/servers/eserver/zseries/software/sa/
www.ibm.com/software/sysmgmt/products/support/
```

How to reach us by e-mail

If you would like to contact us by e-mail, send your comments to eservdoc@de.ibm.com

What's new in this release

Get a quick overview about the new features of System Automation for Multiplatforms version 4.1.1.

Improved operations on the command line with new `samcc` command

System Automation for Multiplatforms version 4.1.0.2 adds a new command `samcc`, which can be used as operations console on the command line interface. For more information, see .

Additional platforms support

System Automation for Multiplatforms version 4.1.0.1 supports the following new platforms:

- SUSE SLES 12 (64-bit)
- Red Hat RHEL 7 (64-bit)
- Ubuntu 14.04 (64-bit): System x, Power Systems (Little Endian only)

System Automation for Multiplatforms version 4.1.0.2 supports the following new platforms:

- Red Hat RHEL 7.1 on Power Systems Little Endian (64-bit)

System Automation for Multiplatforms version 4.1.0.3 supports the following new platforms:

- AIX 7.2

System Automation for Multiplatforms version 4.1.0.4 supports the following new platforms:

- Ubuntu 16.04 (64-bit): System x, Power Systems (Little Endian only).

For more information, see *System Automation for Multiplatforms Installation and Configuration Guide*.

System Automation for Multiplatforms version 4.1.0.5 supports the following new platforms:

- SUSE SLES 15 (64-bit)
- Ubuntu 18.04 (64-bit): System x, Power Systems (Little Endian only).

System Automation for Multiplatforms version 4.1.0.5 adds support for:

- SAP Netweaver 7.5.3 ENSA2.

System Automation for Multiplatforms version 4.1.0.6 supports the following new platforms:

- Red Hat RHEL 8 (64-bit)
- Ubuntu 20.04 (64-bit): System x, Power Systems (Little Endian only)

System Automation for Multiplatforms version 4.1.0.6 adds support for:

- Added SAP NetWeaver support for S/4HANA 1809
- Added SAP NetWeaver support for S/4HANA 1909
- Added support for Oracle 19c
- Added support for SAP HANA 2.0 SPS 04 Revision 046

System Automation for Multiplatforms version 4.1.0.7 supports the following new platforms:

- AIX 7.2 TL5

System Automation for Multiplatforms version 4.1.0.7 adds support for:

- Added SAP NetWeaver support for S/4HANA 2020
- Added support for SAP HANA 2.0 SPS 05 Revision 050

System Automation for Multiplatforms version 4.1.1.0 adds support for:

- New Operating Systems AIX 7.3, RHEL 9 and Ubuntu 22.04 LTS.

System Automation for Multiplatforms version 4.1.1.1 adds support for:

- New Operating Systems AIX 7.2 TL5, RHEL 9.1 and Ubuntu 22.04 LTS.

Improved high availability policy for SAP

The SAP Central Services high availability policy is available as System Automation for Multiplatforms optional feature, which is priced separately. This SAP Central Services high availability policy is now adapted to SAP Netweaver technology.

The user can start and stop the SAP Netweaver stack by using the SAP user interface without interfering with the System Automation policy. The SAP Software Update Manager is able to update the Netweaver solution without the need to disable System Automation during the update process.

Supported SAP configuration options: Java, ABAP, and DUAL stack support for SAP Central Services failover. Additionally, the following configuration options are supported:

- Application server (restart in place of primary and additional application server)
- SAP router failover
- SAP Web Dispatcher failover
- Start after dependency support to database

System Automation for Multiplatforms version 4.1.0.2 adds support for:

- SAP HANA System Replication failover

The supported SAP kernel version is 7.20 or higher.

For more information, see *System Automation for Multiplatforms High Availability Policies Guide*.

Gathering information about application failures

The `samwhy` program is a simple and easy-to-use tool that offers the detection of application failures and their analysis for applications that are controlled by System Automation. `samwhy` helps the operator to understand what happened and provides an explanation why System Automation reacted the way it did.

For more information, see *System Automation for Multiplatforms Reference Guide*.

High availability of the end-to-end automation adapter is simplified

An extra automation policy or virtual IP address is not required any more.

For more information, see System Automation for Multiplatforms Installation and Configuration Guide.

Run the end-to-end automation adapter with a non-root user

By default, the end-to-end automation adapter runs with a root user. Now the adapter can also be set up to run with a non-root user.

For more information, see System Automation for Multiplatforms Installation and Configuration Guide.

Chapter 1. Planning

Planning includes such tasks as assessing your current infrastructure and ensuring that your systems have the required prerequisites.

Planning for installation

Before you install System Automation for Multiplatforms in your AIX and Linux environments, you must ensure that you have the correct prerequisites.

Packaging

System Automation for Multiplatforms can be ordered from IBM® as a media pack or downloaded from an IBM software distribution download site.

Product DVD

Content of the System Automation for Multiplatforms version 4.1 product DVD.

Separate DVDs labeled as follows contain scripts and software packages for each platform and the corresponding architecture:

- Tivoli System Automation for Multiplatforms 4.1 - Linux on System x, Linux on POWER® and Linux on System z
- Tivoli System Automation for Multiplatforms 4.1 - AIX

To install System Automation for Multiplatforms, use the installation script listed in the right column of the table below.

Table 2. Product DVD versions

Operating system	Product DVD label	Installation script
Linux	Tivoli System Automation for Multiplatforms v4.1 - Linux on System x, Linux on POWER and Linux on System z	`SAM4100MPLinux/installSAM`
AIX	Tivoli System Automation for Multiplatforms v4.1 - AIX	`SAM4100MPAIX/installSAM`

Electronic distribution

If you prefer electronic distribution to delivery on the DVD, after purchasing System Automation for Multiplatforms you can download the appropriate archive files from the web by using the supplied URL.

Linux

Table 3. Archive for Linux platforms

Archive name	Description
`SA MP 4.1 Linux.tar`	This is the archive you use to install the product. To extract the archive, GNU tar 1.13 or later is required. Use the **`tar xf`** command to extract the archive. When you have extracted the files, you will find the installation script `installSAM` in the following directory: `SAM4100MPLinux`

AIX

Table 4. Archive for AIX platforms	
Archive name	**Description**
`SA MP 4.1 AIX.tar`	This is the archive you use to install the product. Use the **`tar xf`** command to extract the archive. When you have extracted the files, you will find the installation script `installSAM` in the following directory: `SAM4100MPAIX`

Prerequisites

Make sure that you fulfill the software and hardware requirements for System Automation for Multiplatforms.

Prerequisites on AIX systems

- Root authority is required to install System Automation for Multiplatforms.
- A 32-bit version of Java 7, Java 7.1 or Java 8 is required with the following minimum Service Refresh levels:
  - Java 7.0 SR8: AIX package Java7.jre/Java7.sdk 7.0.0.145
  - Java 7.1 SR2: AIX package Java71.jre/Java71.sdk 7.1.0.25
  - Java 8.0 SR0: AIX package Java8.jre/Java8.sdk 8.0.0.507
  - System Automation for Multiplatforms Fixpack Version 4.1.0.7 supports Java 8 SR7 FP16

Prerequisites on Linux systems

The following prerequisites must be met before System Automation for Multiplatforms can be installed on a Linux system:

- The following package is required on each RedHat v7.1 system:
  - `perl-Sys-Syslog`
- The following package is required on each RedHat v8 system:
  - `perl-Net-Ping`
  - `perl-Thread-Queue`
- The following packages are required on each RHEL (7/8) and SLES (12/15) system:
  - `mksh`
  - `psmisc`
- Root authority is required to install System Automation for Multiplatforms.

RSCT packages

During installation of System Automation for Multiplatforms on AIX, the levels of RSCT packages that are required by System Automation for Multiplatforms are checked against the levels of RSCT packages already installed with the operating system, and missing packages or higher levels of RSCT packages are installed if required. Under certain circumstances, you might need to manually install higher levels of certain RSCT packages. For example, if the RSCT basic package is not installed, and the level of the installed RSCT core package is higher than the level of the RSCT packages, which is supplied with System Automation for Multiplatforms, the installation of the RSCT basic package can fail due to RSCT prerequisites not being met. You need to download and install the appropriate RSCT file sets from the AIX service center to ensure that all RSCT packages installed are at the same level.

System Automation for Multiplatforms Version 4.1.0.0 includes RSCT level 3.1.5.3 (APAR IV52893).

System Automation for Multiplatforms Version 4.1.0.7 includes RSCT level 3.2.6.2 (Linux 64bit OS) and RSCT level 3.2.6.1 (AIX OS).

Requirements for virtual environments like KVM or VMWare

Because virtual machines often do not have a reliable way to keep track of time, CPUs with Time Stamp Counter are susceptible to synchronization issues. To avoid time synchronization issues, configure an appropriate time synchronization, for example NTP, for nodes that are running in virtual environments.

Checking prerequisites

Find out how to run a prerequisites check.

Complete the following steps:

1. Log in as root, or with equivalent authority.
2. If you downloaded the tar file from the Internet, extract the file:

   ```
   tar -xvf <tar file>
   ```

   If you received the product on a DVD, mount the DVD and change to the directory where the DVD is mounted.
3. Enter the following command:

   - **Linux: `cd SAM4100MPLinux`**
   - **AIX: `cd SAM4100MPAIX`**

   For information about supported platforms, see "Supported platforms" on page 4
4. To start the prerequisites check, issue the following command:

   ```
   ./prereqSAM
   ```

   Typically, you do not specify any of the options that are available for the **`prereqSAM`** command. For a detailed description of the command, see *Tivoli System Automation for Multiplatforms Reference Guide* .
5. When the check is complete, check the following log file for information about missing prerequisites:

   ```
   /tmp/prereqSAM.<#>.log
   ```

   The <*#*> tag is a number; the highest number identifies the most recent log file.
6. If your system did not pass the prerequisites check, correct any problems before you start the installation.

Installation prerequisites

Before you start the installation, you must fulfill these requirements:

- You need to have root authority to install System Automation for Multiplatforms on the system.
- A Korn shell must be installed on all OS platforms except the RHEL and SUSE OS platforms where a MirBSD Korn Shell (mksh) must be installed.
- Perl is required to use the command-line interface of System Automation for Multiplatforms including native RSCT commands. The command-line interface is installed by default on your Linux or AIX systems as part of the operating system. If you are using System Automation for Multiplatforms in a language other than English, a special version of Perl might be required. Due to known problems with Perl 5.8.0 and how it handles UTF-8 encoded locales, some characters might not be properly displayed. The problem can occur on systems with Perl 5.8.0 is installed, if you use a UTF-8 encoded locale. When previous or subsequent versions of Perl are used, or non-UTF-8 encoded locales are used, this problem does not occur.

  If you decide to upgrade your Perl 5.8.0 version on a Linux distribution, process the following steps:

1. Download the Perl 5.8.1 source.
2. Extract the file on any directory by using **-xvf**.
3. Compile and install on the UTF-8 system, referring to the instructions provided with the downloaded files.
4. Change the symbolic link, which is pointing to the directory of the Perl version that is used by System Automation for Multiplatforms

   Change link from:

   ```
   /usr/sbin/rsct/perl5/bin/perl->/usr/bin/perl
   ```

   To the directory where the new version of Perl is installed:

   ```
   /usr/sbin/rsct/perl5/bin/perl->/usr/local/bin/perl
   ```

- Make sure that the directories `/usr/sbin` and `/opt` have at least 100 MB free space, and that the directory `/var` also provides at least 100 MB free space.
- On any node where the end-to-end automation adapter is configured to run, at least 128 MB RAM must be available.
- During installation of System Automation for Multiplatforms on AIX, the levels of RSCT packages that are required by System Automation for Multiplatforms are checked against the levels of RSCT packages already installed with the operating system. Missing packages or higher levels of RSCT packages are installed if required. Under certain circumstances, you might need to manually install higher levels of certain RSCT packages. For example, if the RSCT basic package is not installed, and the level of the installed RSCT core package is higher than the level of the RSCT packages that are supplied with System Automation for Multiplatforms, the installation of the RSCT basic package might fail due to RSCT prerequisites not being met. You need to download and install the appropriate RSCT file sets from the AIX service center to ensure that all RSCT packages installed are at the same level.
- For other operating system-specific requirements, see the Software Product Compatibility Reports.
- For languages that are using the double-byte character set (DBCS), the Telnet dialog buffer must be sufficiently large to ensure that long messages are properly displayed. Otherwise, enlarge the Telnet dialog buffer.
- In some RHEL distributions, the SELinux environment is turned on by default. Make sure that the SELinux environment is switched off so that System Automation for Multiplatforms works properly.
- As per SUSE announcement, the support for Korn Shell (Ksh) will become EoS after Q1’22 on SUSE distributions.

Supported platforms

Find out which platforms are supported by System Automation for Multiplatforms.

System Automation for Multiplatforms supports the following UNIX environments:

- Linux on System z
- Linux on System x
- Linux on Power®
- Ubuntu on System x[5]
- Ubuntu on Power[5]
- AIX

System Automation for Multiplatforms runs on:

- All IBM Systems machines running Linux.
- IBM System p machines running AIX.

System Automation for Multiplatforms runs under:

- VMware on IBM System x (except Intel IA64-based servers) and any other 32-bit Intel-based server, AMD Opteron-based server (64-bit), or Intel EM64T-based server (64 bit). Live migration of systems using vMotion is supported (see “VMware vMotion support” on page 6).
- RHEV-H version 4.3, KVM hypervisor version 5.4 or higher on all supported Linux distributions on IBM System x. Live migration of systems is not supported.

The following table lists the supported operating system versions.

www.ibm.com/software/tivoli/products/sys-auto-multi/

Table 5. Supported UNIX and Linux platforms of System Automation for Multiplatforms

OS Platform	IBM System x[1]	IBM System z	Power Systems	Power Systems (Little Endian)
SUSE SLES 15.5	x	x		x
Red Hat RHEL 9	x	x		x
Red Hat RHEL 8	x	x		x
Ubuntu 22.04 LTS	x			x
Ubuntu 20.04 LTS	x			x
AIX 7.2			x	
AIX 7.3			x	

All SP levels of above listed supported SUSE versions and Red Hat versions are also supported, unless one of the following notes indicates a more specific minimum requirement.

Note:

1. System x means System x (except Intel IA64 based servers) and any other 32-bit Intel based server, or AMD Opteron based server (64-bit), or Intel EM64T based server (64 bit).
2. zSystems version z15 and pSystems version p9 are supported.
3. All future/new SP levels of Linux (SUSE & RHEL) are supported, if they are supported by the RSCT packages bundled (see “Prerequisites” on page 2 for more details) with this fix pack and are backward compatible with the SP level qualified with this fix pack.
4. Platform support is introduced with fix pack 4.1.0.1.
5. Platform support is introduced with fix pack 4.1.0.2.
6. Platform support is introduced with fix pack 4.1.0.4.
7. Platform support is introduced with fix pack 4.1.0.5.
8. Platform support is introduced with fix pack 4.1.0.6.
9. Platform support is introduced with fix pack 4.1.0.7.

For more information, see “Installing on new operating systems” on page 32 on page 34.

Supported network interfaces

All platforms support 10 Megabit Ethernet, Fast Ethernet, and Gigabit Ethernet. In addition, the System z platform also supports HiperSockets, CTC, and VM Guest LAN.

Support for network file systems

System Automation for Multiplatforms supports network file systems on Linux on POWER, Linux on System x, Linux on System z, and AIX.

Network file systems are not harvested. To automate a network file system, you use user-defined `IBM.AgFileSystem` resources.

Restriction:

- Network file systems of class `IBM.AgFileSystem` can be automated and monitored successfully only if the `root` user of the importing system has write access to the file system.
- Cascaded usage of file systems is not possible:

  With System Automation for Multiplatforms you can define a highly available NFS server, where the exported file systems are automated as resources of class `IBM.AgFileSystem` which reside on a shared disk medium. The NFS server itself is automated as a resource of class `IBM.Application` which may float on systems that have access to the shared disk medium. When an additional system imports the network file systems, however, the imported file systems must not already exist as user-defined `IBM.AgFileSystem` resources on the importing system, otherwise, monitoring of the file systems fails and the resources go into OpState 3 (`FAILED OFFLINE`).

Live Partition Mobility support requirements

With AIX Level 6100-00-01 (or higher) installed on the source and destination POWER6® servers, the Live Partition Mobility feature can be used to migrate an LPAR running as a System Automation for Multiplatforms node. The state or operation of the System Automation for Multiplatforms cluster is not affected. The cluster is configured to use standard (default) heartbeat settings. In that case, the effect on the application servers is a brief interruption of operations during the migration. You don't need to restart System Automation for Multiplatforms or the application servers.

Make sure that the period of interruption during Live Partition Mobility does not cause unwanted cluster events. Unwanted cluster events occur if too many heartbeats from the node are missed during the average period of interruption. In that case, relax the heartbeat settings for the time of Live Partition Mobility.

Another way to minimize the chance of unwanted cluster events while an LPAR is moved is to stop the peer domain forcefully before the move is initiated with `stoprpdomain -f`, that is, without stopping the applications managed by the cluster services. After the move is completed, restart the peer domain.

Restriction: Disk tiebreaker is not supported by virtual SCSI which is a prerequisite of Live Partition Mobility.

VMware vMotion support

With a VMware vSphere setup with multiple ESX servers managed by a vCenter Server, the vMotion feature can be used to migrate live guests running as an System Automation for Multiplatforms node. The migration does not affect the state or operation of the System Automation for Multiplatforms cluster, provided that the cluster is configured to use standard (that is, default) heartbeat settings. In that case, the effect on the application servers running under System Automation for Multiplatforms control is a brief interruption of operations during the migration. Neither System Automation for Multiplatforms nor the application servers will have to be restarted.

Make sure that the period of interruption during vMotion does not cause unwanted cluster events. Unwanted cluster events will occur if too many heartbeats from the node are missed during the average period of interruption. In that case, relax the heartbeat settings for the time of vMotion.

Another way to minimize the chance of unwanted cluster events while an virtual guest is moved is to stop the peer domain forcefully before the move is initiated with `stoprpdomain -f`, that is, without stopping the applications managed by the cluster services. After the move is completed, restart the peer domain.

System Automation for Multiplatforms supports vMotion for ESX and ESXi servers with version 3.5 or higher and the following guest operating systems:

- SLES 12 or 15 (x86-64)
- RHEL 7 or 8 (x86-64)
- Ubuntu 18.04 or 20.04 (x86-64)

Limitations: System Automation for Multiplatforms does not support vMotion of nodes that use shared storage, because vMotion does not support shared real or virtual storage volumes (disks).

z/VM single system image and live guest relocation support

z/VM 6.2 introduces Single System Image (SSI) support, which is a multi-system clustering technology. You can cluster up to 4 z/VM images by using Single System Image. SSI facilitates resource sharing among the members in the cluster. You can move an active Linux on System z guest to another z/VM system without an outage of the guest. This facility is called Live Guest Relocation (LGR) and is only supported for Linux on System z guests.

To understand the z/VM SSI and LGR concepts and capabilities, refer to An Introduction to z/VM Single System Image (SSI) and Live Guest Relocation (LGR) (SG24-8006).

If the requisite level of z/VM is installed on the source and destination systems, the Live Guest Relocation feature of z/VM can be used to relocate a z/VM Linux guest system. If the requisite level of System Automation for Multiplatforms is installed on the Linux guest system, relocating this guest system does not affect the state or operation of the System Automation for Multiplatforms cluster if standard (default) heartbeat settings are configured. For an application that is managed by System Automation for Multiplatforms, the process of relocation is a brief suspension of operations. Restart is not required for System Automation for Multiplatforms and the application.

Validate that the period of suspension during Live Guest Relocation does not cause unwanted cluster events. Unwanted cluster events occur if a configured number of heartbeats is missing from the node that experiences a suspend during Live Guest Relocation. If testing shows that the average period of suspension can cause too many heartbeats to be missed, the heartbeat settings should be relaxed for the time of Live Guest Relocation. To greatly minimize the chance of unwanted cluster events while a z/VM guest system is relocated, stop the peer domain forcefully before the relocation is initiated by using **`stoprpdomain -f`**. For example, without stopping the applications that are managed by the cluster services. Upon successful completion of the relocation, restart the peer domain by using the **`startrpdomain`** command.

Requirements

- System Automation for Multiplatforms version 3.2.2.4 (or higher)
- z/VM version 6.2

Limitations

ECKD Disk tiebreaker and SCSI PR tiebreaker cannot be used with Live Guest Relocation, because guests that hold a reserve on a disk cannot be relocated.

Preparing for installation

System Automation for Multiplatforms is contained in several packages that must be installed on every cluster node that you want to automate. Package type and content depend on the operating system on which you are installing System Automation for Multiplatforms.

Starting the configuration

Execute the following initial configurations:

- On all nodes, set and export the environment variable CT_MANAGEMENT_SCOPE to 2 (peer domain scope) for all users of System Automation for Multiplatforms : `export CT_MANAGEMENT_SCOPE=2`

  To permanently set the variable, set and export it in the profile.

  On SLES systems, you can create scripts in `/etc/profile.d` with the following content:

  ```
  sa_mp.sh:
  export CT_MANAGEMENT_SCOPE=2
  sap_mp.csh :
  setenv CT_MANAGEMENT_SCOPE 2
  ```

- Make sure that the environment variable LANG is set to one of the supported locales for the `root` user. To set the environment variable, use the command:

  ```
  export LANG=xx_XX
  ```

  xx_XX denotes one of the supported languages.

For a list of supported languages and locales, see “Supported languages and locales” on page 21.

Load on nodes

System Automation for Multiplatforms requires some of its subsystems to be processed constantly on the node to ensure that the cluster services are working properly (for example, heartbeat and communication between the subsystems). If this is not possible, System Automation might trigger critical resource protection methods in case those subsystems cannot communicate within a short period of time. This protection mechanism eventually leads to a restart of the node on which this issue occurs.

To prevent an unwanted system restart, constant I/O and swap load must be lower than 10%.

For more information about critical resource protection methods, see *Tivoli System Automation for Multiplatforms Administrator's and User's Guide*.

Number of nodes in a cluster

Linux
The maximum number of nodes in a cluster is 32.

AIX
The maximum number of nodes in a cluster is 130.

Note:

1. The software packages must be available on the nodes on which you want to install System Automation for Multiplatforms. For example, you can mount the DVD on a PC and use FTP to transfer the files to the node, or you can install the packages over a shared Network File System.
2. To be sure that the software packages are installed and uninstalled correctly, use the System Automation for Multiplatforms scripts **`installSAM`** and **`uninstallSAM`**. They also perform requirements checking, license installation and migration tasks.
3. With the exception of the language packages, all packages are required for System Automation to work. Starting with System Automation for Multiplatforms 4.1 it is no longer possible to uninstall the RSCT package `rsct.opt.storagerm` without uninstalling the entire product.

Planning for new platform support

Starting with fix pack 4.1.0.7, System Automation for Multiplatforms Linux32 package has been discontinued as RHEL6 and SLES11 have become end of support.

Following packages are provided for all fix packs 4.1.1.x. Both packages have the same code basis.

- • The first package `4.1.1-TIV-SAMP-AIX-FPxxxx` contains the System Automation for Multiplatforms product build for AIX environments. These environments are required by System Automation for Multiplatforms under the AIX 7 operating systems.

- • The second package `4.1.1-TIV-SAMP-Linux64-FPxxxx` contains the System Automation for Multiplatforms product build for 64-bit language environments. These 64-bit language environments are required by System Automation for Multiplatforms under the Linux operating systems RHEL 7/8, SLES 12/15 and Ubuntu 18.04/20.04.

System Automation for Multiplatforms 4.1.0.0 or lower is not supported.

Planning for a highly available network infrastructure

Understand the complexity and plan the setup of a highly available network.

The following figure shows a network infrastructure on Linux.

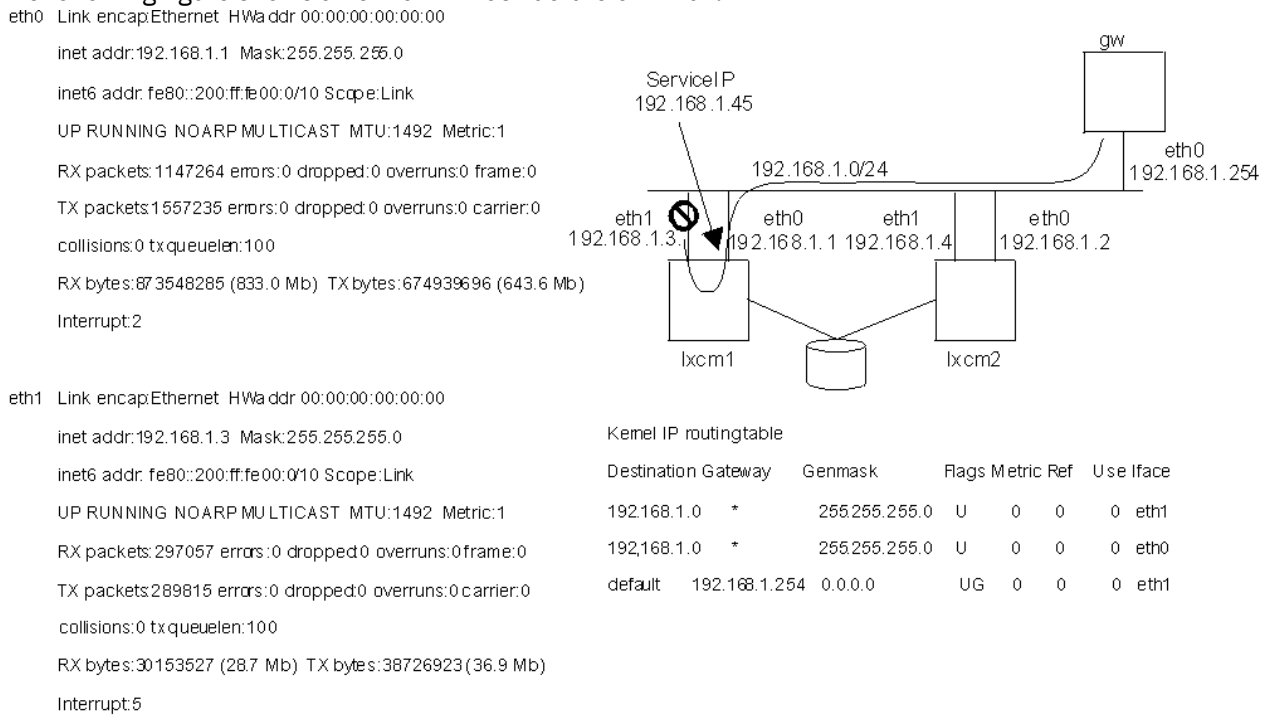

Figure 2. Problems planning a highly available network

Every configured static network device is identified by an entry in the routing table. The routing algorithm chooses the first matching route out of this table. In this example, device `eth1` on node `lnxcm1` fails. As `eth1` is the first entry in the routing table, the node cannot send packages out to the network although there is another working network interface `eth0`.

Consider the following questions before you start to plan your high availability network:

1. What kind of high availability network do you need?
   - Is it required to move a `ServiceIP` from one interface to another on the same node?
   - Is it required to switch to another node which has a working interface in the required sub-net?
2. Can you implement additional IP sub-nets or do you use an existing network infrastructure?
3. Do you work only in the scope of our cluster nodes or are you able to implement or deploy network services on other nodes outside of the automation cluster?
4. What network hardware do you have?

Depending on how you answered the questions, you may want to choose one of the following setups to develop your own high availability network strategy.

Planning for storage devices

Using single-path storage devices

Support for single-path storage devices is different depending on your operating environment.

AIX

Full support is provided for single-path storage devices:

- Harvested `IBM.AgFileSystem` resources can be automated.

  `IBM.AgFileSystem` resources are harvested if they are of type `jfs` or `jfs2` and reside on storage entities that are harvested themselves (storage entities of class `IBM.LogicalVolume`, `IBM.VolumeGroup`, `IBM.Disk`.
- User-defined `IBM.AgFileSystem` resources can be automated, for example, network file systems.
- SCSI-2 reservation is supported.

Limitations:

- No striping
- User-defined `IBM.AgFileSystem` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.

Linux on POWER and Linux on System x

Limited support is provided:

- Harvested `IBM.AgFileSystem` resources can be automated.

  `IBM.AgFileSystem` resources are harvested if they are of type `ext2`, `ext3`, or `reiserfs` and reside on storage entities that are harvested themselves (storage entities of class `IBM.LogicalVolume`, `IBM.Partition`, `IBM.VolumeGroup`, `IBM.Disk`.
- User-defined `IBM.AgFileSystem` resources can be automated, for example network file systems.

Limitations:

- Support for SCSI reservation is limited. Perform a disk reserve operation to check whether SCSI reservation is available.
- User-defined `IBM.AgFileSystem` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.

Linux on System z

Device mapper provided or md devices are harvested as `IBM.Disk` resources if a physical volume has been created on the md device using the **`pvcreate`** command.

Limitations:

- Only user-defined `IBM.AgFileSystem` resources or `IBM.AgFileSystem` resources residing on harvested device mapper provided or md devices can be automated. Resource harvesting for other disks is not supported. Even if harvesting of other disk resources is successful, the harvested resources cannot be automated.
- User-defined `IBM.AgFileSystem` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.
- SCSI reservation is not supported.

Using multipath storage devices

Depending on your environment, support for multipath storage devices might have some restrictions.

AIX

Full support is available for SPIO and MPIO storage devices:

- Harvested `IBM.AgFileSystem` resources can be automated.

  `IBM.AgFileSystem` resources are harvested if they are of type `jfs` or `jfs2` and reside on storage entities that are harvested themselves (storage entities of class `IBM.LogicalVolume`, `IBM.VolumeGroup`, `IBM.Disk`).
- User-defined `IBM.AgFileSystem` resources can be automated (for example, network file systems).
- SCSI-2 reservation is supported for SPIO and MPIO storage devices using the Redundant Disk Array Controller (RDAC) driver.

Note: This driver is only available for the IBM TotalStorage DS4k and DS6k families.

Limitations:

- No striping
- User-defined `IBM.AgFileSystem` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.

Linux on POWER and Linux on System x

Full support is available for single-path I/O (SPIO) storage devices, and for multipath storage I/O (MPIO) devices with Redundant Disk Array Controller (RDAC) device drivers, as well as md and device mapper provided devices.

- Harvested `IBM.AgFileSystem` resources can be automated.

  `IBM.AgFileSystem` resources are harvested if they are of type `ext2`, `ext3`, or `reiserfs` and reside on storage entities that are themselves harvested (storage entities of class `IBM.LogicalVolume`, `IBM.Partition`, `IBM.VolumeGroup`, `IBM.Disk`.
- User-defined `IBM.AgFileSystem` resources can be automated (for example, network file systems).
- SCSI-2 reservation is supported for disks harvested from the RDAC driver.
- Linux RAID (`/dev/device mapper` provided or `md` devices) is supported.
- Device mapper managed disks are supported.

Limitations:

- File systems created on device mapper provided or md devices without using LVM are not harvested, they can only be automated using user-defined `IBM.AgFileSystem` resources.
- device mapper provided or md devices themselves are only harvested as `IBM.Disk` resources if a physical volume has been created on the md device using the **`pvcreate`** command.
- SCSI-2 reservation is not supported for non-RDAC drivers or for device mapper provided or md devices themselves.
- User-defined `IBM.AgFileSystem` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.
- EVMS is not supported, which includes any Volume Groups/Logical Volumes created or managed by EVMS.
- For SLES 12/15 and RHEL 7/8, harvesting of storage entities of class IBM.Disk, IBM.VolumeGroup, IBM.LogicalVolume, IBM.Partition, and IBM.AgFileSystem are supported. File systems may be automated if the limitations for device mapper provided or md devices listed above are met.

Linux on System z

Device mapper provided or md devices are harvested as `IBM.Disk` resources if a physical volume has been created on the provided block device using the `pvcreate` command. This is independent of the underlying disk technology, ECKD or SCSI.

Limitations:

- Only user-defined `IBM.AgFileSystems` resources or `IBM.AgFileSystems` resources residing on harvested device mapper provided or md devices can be automated. Resource harvesting for other disks is not supported. Even if harvesting of other disk resources is successful, the harvested resources cannot be automated.
- User-defined `IBM.AgFileSystems` resources can only be automated if the disk hosting the file system has the same device name on all nodes of the cluster.
- SCSI reservation is not supported.

Using network interfaces

You can set up a high availability configuration with two nodes in a cluster, each with two network interfaces.

Before you start with this setup, keep in mind that it is not possible to have more than one static configured network interface in the same IP subnet. Each IP address causes an entry in the kernel routing table. If threre are two interfaces in the same subnet, there are 2 routes for the same subnet. If the interface, which created the first entry fails, the communication for this subnet breaks down even if there is another interface, which still is able to communicate.

Two physically separated networks, move ServiceIP between nodes

The following network setup applies:

Table 6. Network setup for a two-node cluster with network interfaces

Resource	Name	Device	IP
Cluster node	lnxcm1	eth0 eth1	9.152.172.1/24 192.168.1.1/24
Cluster node	lnxcm2	eth0 eth1	9.152.172.2/24 192.168.1.2/24
Router	gw	eth0	9.152.172.254/24
ServiceIP	-	–	9.152.172.3/24

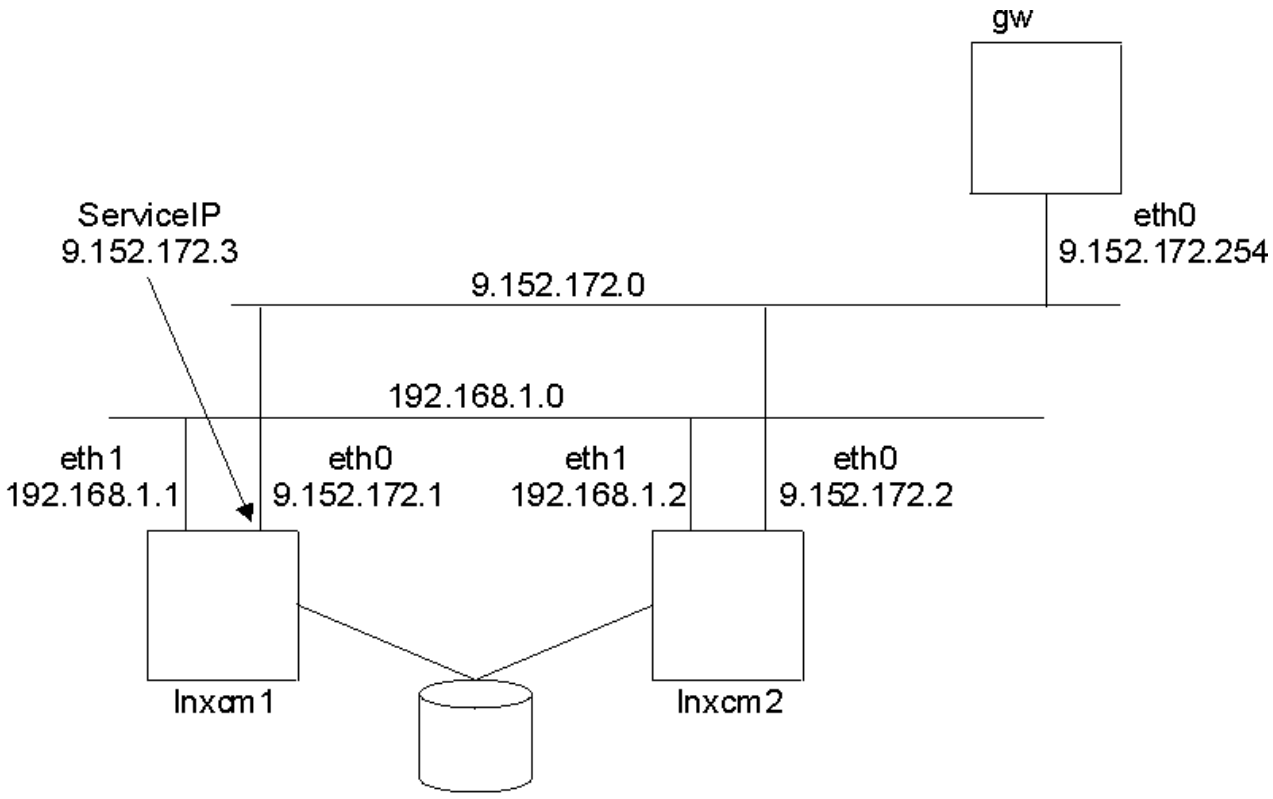

Figure 3. Two nodes, two interfaces, two physically separated networks

There are now two networks 192.168.1.0 and 9.152.172.0 for the cluster communication. If there is a failure in one network interface the cluster will not break.

- Network 9.152.172.0 represents the network for the highly available IT service.
- Network 192.168.1.0 makes cluster internal communication more reliable.

Since only the network of the ServiceIP is connected to the gateway, a failure of interface eth0 on lnxcm1 will cause the automation to move the ServiceIP to the interface eth0 on the other node lnxcm2. Because of the physical separation of the two networks it is not possible to move the ServiceIP from eth0 to eth1 within the same node.

The sample System Automation for Multiplatforms policy is the same as shown in Figure 7 on page 17.

Table 7. Advantages and disadvantages of a two-node setup with network interfaces

Advantage	**Disadvantage**
Easy setup.	ServiceIP moves only between nodes.
Redundancy in cluster communication.	

Three logical networks in one physical network, move `ServiceIP` between network interfaces

Another network setup is required to not only move the `ServiceIP` between nodes in the cluster but also between interfaces within one node.

A separate logical network for each interface of a node is required, and an additional network for the `ServiceIP`. Choosing an existing network (one of eth0 or eth1) can cause routing problems. Make sure to connect all interfaces to the same physical network. This allows each interface to hold addresses of all the logical networks.

The following network setup applies:

Table 8. Network setup for three logical networks in one physical network			
Resource	**Name**	**Device**	**IP**
Cluster node	lnxcm1	eth0 eth1	192.168.1.1/24 192.168.2.1/24
Cluster node	lnxcm2	eth0 eth1	192.168.1.2/24 192.168.2.2/24
Router	gw	eth0	9.152.172.254/24
ServiceIP	-	–	9.152.172.3/24

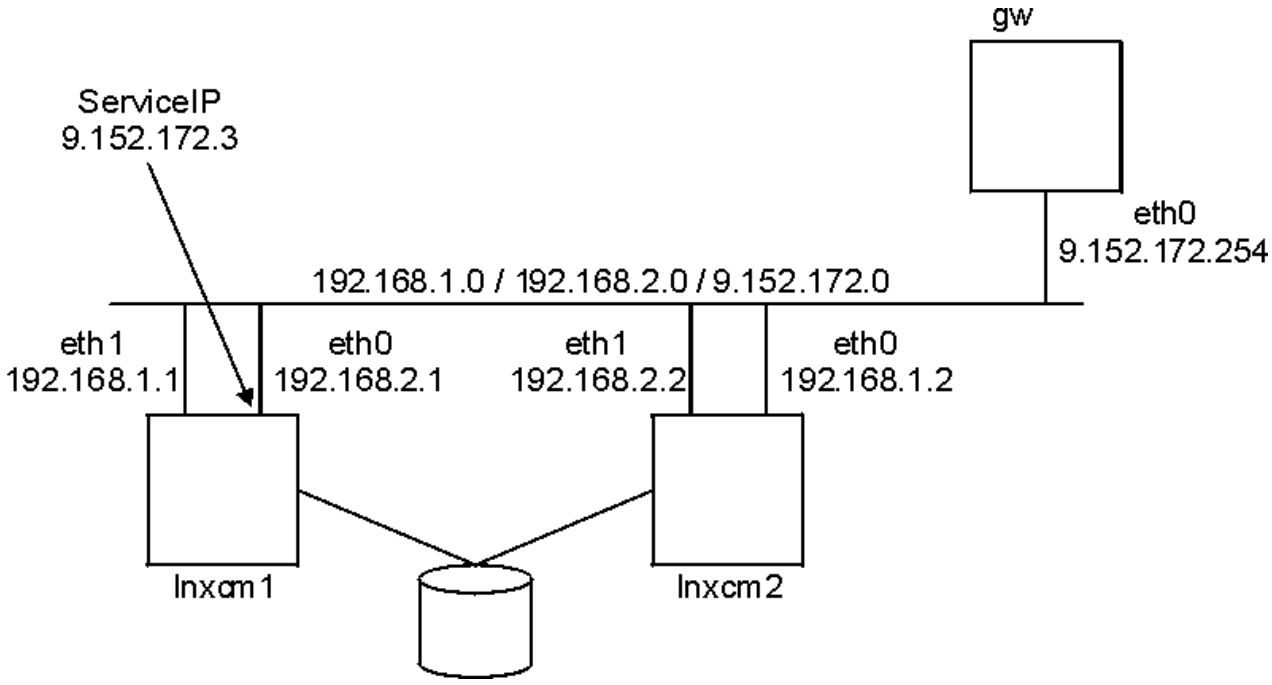

Figure 4. Two nodes, two interfaces, one physical network

- Network 9.152.172.0 represents the network for the highly available IT service.
- Network 192.168.1.0 represents the first cluster internal communication network.
- Network 192.168.2.0 represents the second cluster internal communication network.

Sample System Automation for Multiplatforms policy:

```
lnxcm1# mkequ NetInt
IBM.NetworkInterface:eth0:lnxcm1,eth1:lnxcm1,eth0:lnxcm2,eth1:lnxcm2
lnxcm1# mkrsrc IBM.ServiceIP Name="SIP" IPAddress="9.152.172.3"
NetMask="255.255.255.0" NodeNameList="{'lnxcm1','lnxcm2'}"
lnxcm1# mkrg rg
lnxcm1# addrgmbr -g rg IBM.ServiceIP:SIP
lnxcm1# mkrel -p dependson -S IBM.ServiceIP:SIP -G IBM.Equivalency:NetInt
```

Table 9. Advantages and disadvantages of a network setup for three logical networks in one physical network	
Advantage	**Disadvantage**
Easy setup.	3 logical networks in 1 physical network.
Redundancy in cluster communication.	Traffic of 3 networks on 1 physical medium.

Table 9. Advantages and disadvantages of a network setup for three logical networks in one physical network (continued)	
Advantage	**Disadvantage**
ServiceIP can move between interfaces and nodes.	

Two physically separated networks, dynamic routing and VIPA

A detailed description of this setup is beyond the scope of this manual. Basically the ServiceIP is assigned to a virtual network within the kernel of a cluster node. Dynamic routing on all cluster nodes and the gateway makes sure that a route to the ServiceIP is established.

The following network setup applies:

Table 10. Network setup of two physically separated networks			
Resource	**Name**	**Device**	**IP**
Cluster node	lnxcm1	eth0 eth1	9.152.170.1/24 9.152.171.1/24
Cluster node	lnxcm2	eth0 eth1	9.152.170.2/24 9.152.171.2/24
Router	gw	eth0 eth1	9.152.170.254/24 9.152.171.254/24
ServiceIP	-	–	9.152.172.3/24

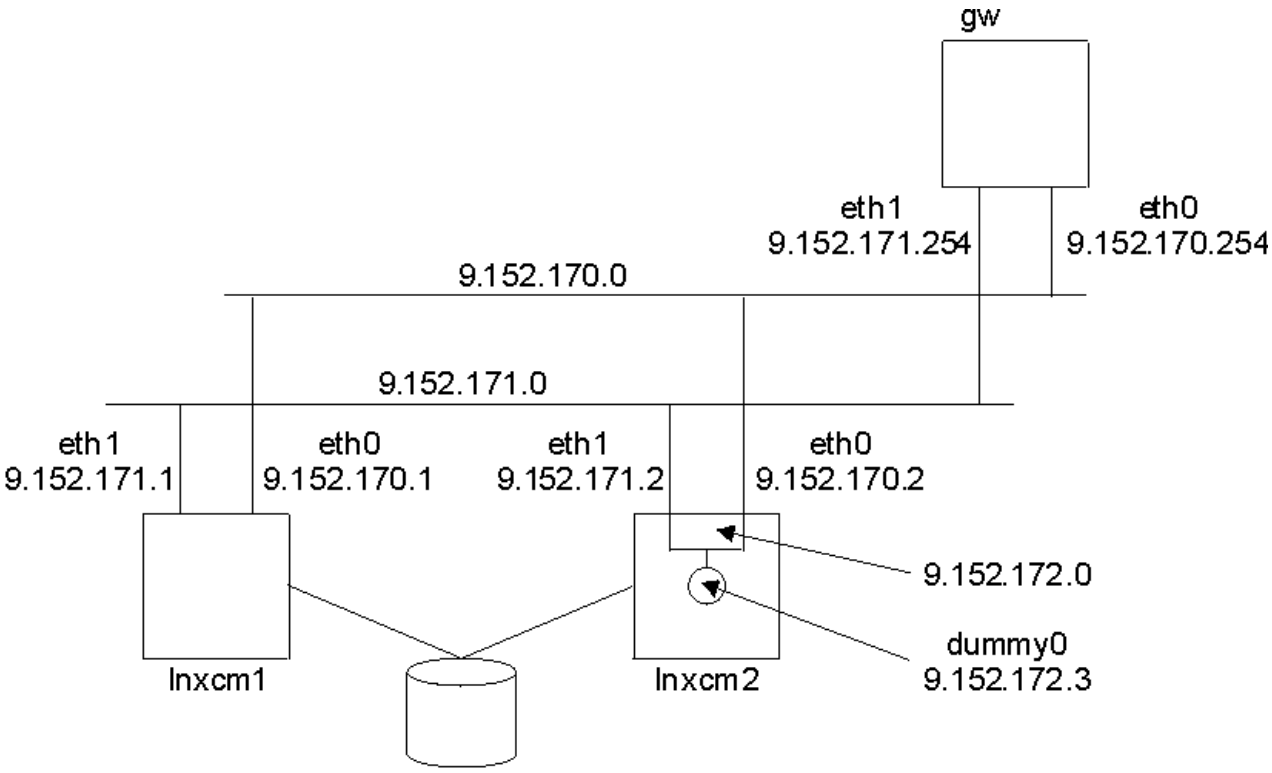

Figure 5. Two physically separated networks, dynamic routing and VIPA

Table 11. Advantages and disadvantages for a network setup of two physically separated networks	
Advantage	**Disadvantage**
There is no dependency to the physical network device.	Complicate setup.
Concept of finding dynamically the best way to a host (IP address).	Dynamic routing required.
No need to move `ServiceIP` between interfaces.	Setup is not restricted to the cluster nodes; gateway also has to support dynamic routing.

Interface bonding

Several physical network interfaces are bonded together to one logical network device. The operating system has to support this feature with a special bonding device driver. Consult your operating system documentation how to configure interface bonding on your system. Make sure that you configure high availability (high availability) bonding and ensure your network interface cards support the interface failure detection mechanism your bonding driver requires.

The following network setup applies:

Table 12. Network setup for physical network interfaces that are bonded together			
Resource	**Name**	**Device**	**IP**
Cluster node	lnxcm1	eth0 eth1	9.152.172.1/24 9.152.172.1/24
Cluster node	lnxcm2	eth0 eth1	9.152.172.2/24 9.152.172.2/24
Router	gw	eth0	9.152.172.254/24
ServiceIP	-	-	9.152.172.3/24

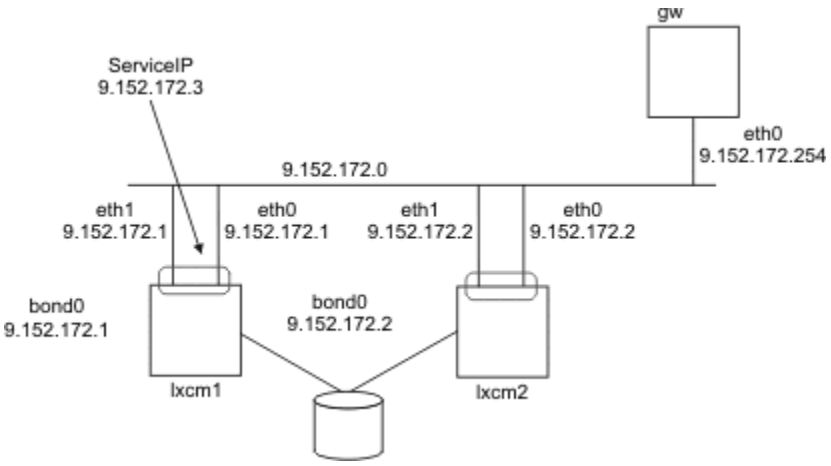

Figure 6. Network interfaces bonded together to one logical network device

Table 13. Advantages and disadvantages for a network setup for physical network interfaces that are bonded together	
Advantage	**Disadvantage**
Easy setup.	Operating system has to support interface bonding.
Redundancy in cluster communication.	Network interface hardware may has to support interface failure detection (for example, MII link monitoring).
There is no need to move ServiceIP between devices on the same node.	

Using an Ethernet interface

You can set up a high availability configuration with two nodes in a cluster each with a separate Ethernet interface.

The following network setup is given:

Table 14. Network setup of a two-node cluster with Ethernet interfaces			
Resource	**Name**	**Device**	**IP**
Cluster node	lnxcm1	eth0	9.152.172.1/24
Cluster node	lnxcm2	eth0	9.152.172.2/24
Router	gw	eth0	9.152.172.254/24
ServiceIP	–	–	9.152.172.3/24

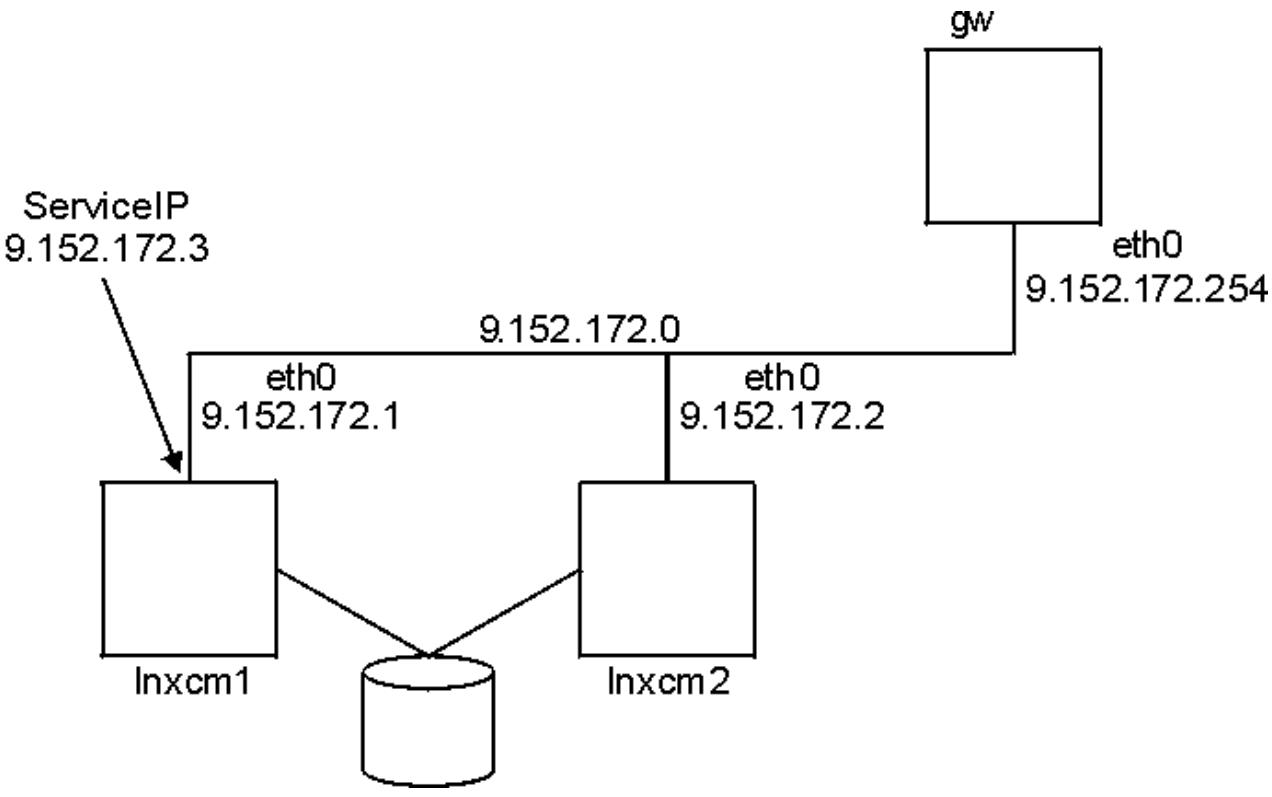

Figure 7. Two nodes, one interface

In this setup, the cluster communication and the presentation of the highly available IT service use the same communication path, the 9.152.172.0 network.

Automation can assign the ServiceIP either on the lnxcm1 interface eth0 or on the lnxcm2 interface eth0. If one interface fails, automation moves the ServiceIP to the other node. Thus it satisfies the policy that requires assigning the ServiceIP on a running network interface.

In this setup the failure of one network interface leads to a break in the cluster communication with all the problems described in System Automation for Multiplatforms Administrator's and User's Guide. If the communication breaks as shown in Figure 8 on page 18, the tiebreaker decides which node continues with the automation. If the tiebreaker is reserved by node lnxcm1, then no online network interface is available on node lnxcm1 to assign the ServiceIP.

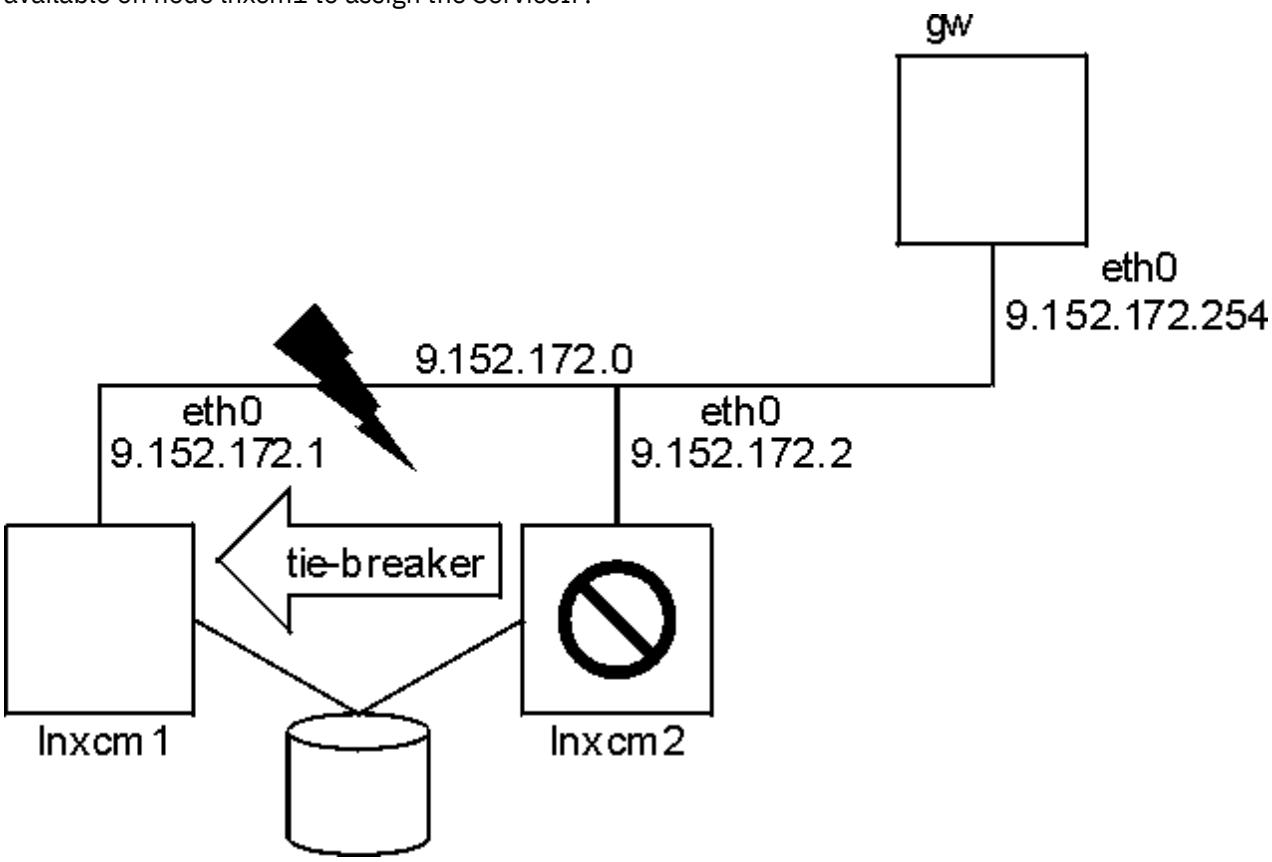

Figure 8. Two nodes, one interface – interface failure

In this example the network 9.152.172.0 served two purposes:

1. Representing the network for the highly available IT service.
2. Used for internal cluster communication.

Sample System Automation for Multiplatforms policy:

```
lnxcm1# mkequ NetInt IBM.NetworkInterface:eth0:lnxcm1,eth0:lnxcm2
lnxcm1# mkrsrc IBM.ServiceIP Name="SIP"
IPAddress="9.152.172.3"
NetMask="255.255.255.0"
NodeNameList="{'lnxcm1','lnxcm2'}"
lnxcm1# mkrg rg
lnxcm1# addrgmbr -g rg IBM.ServiceIP:SIP
lnxcm1# mkrel -p dependson -S IBM.ServiceIP:SIP -G IBM.Equivalency:NetInt
```

Table 15. Advantages and disadvantages of a two-node cluster with Ethernet interfaces

Advantage	**Disadvantage**
The set up is simple.	Each communication problem leads to cluster split.
Less network hardware required.	ServiceIP moves only between nodes.

Chapter 2. Installing

Installing or upgrading System Automation for Multiplatforms involves preparing your system and running a set of tasks that are specific to your environment.

Upgrading

You can upgrade System Automation for Multiplatforms either from a Try & Buy version to a full version or from a running version to the newest release.

Upgrading from a Try & Buy version to a full product version

The Try & Buy version of System Automation for Multiplatforms is installed and you purchased the full product version. Then, you receive another copy of the installation media, which contains the license file for the full license.

The license file is on the installation medium in the `license` subdirectory. To run the license upgrade, enter:

```
samlicm -i <license_file_name>
```

To display the license, enter:

```
samlicm -s
```

After you upgraded the license, check if any updates for System Automation for Multiplatforms are available and install the updates.

Upgrading from a version earlier than version 4.1

You can upgrade to version 4.1 from earlier versions of the product.

When you upgrade System Automation for Multiplatforms from a version earlier than version 4.1, observe the following comments:

Silent adapter configuration
If you are using the `cfgsamadapter` configuration utility in silent mode to configure the end-to-end automation adapter settings, make sure that you generate a new silent input properties file on the new release level. The automation adapter settings are configured in silent mode when you start the `cfgsamadapter` utility by using the option `-s`. Before you can run any silent configuration, generate a new input properties file by opening the `cfgsamadapter` utility with the options `-s [ -g | -gr ]` rather than using an existing input properties file.

Removed operations console
The operations console and the policy editor are not contained in version 4.1. You can still use the operations console to operate first-level domains and the policy editor that is provided by System Automation for Multiplatforms up to version 3.2.2 to maintain policies.

Installing System Automation for Multiplatforms

You can install System Automation for Multiplatforms in your environment, or you can upgrade a previous version of the product.

The following topics explain how to install or upgrade System Automation for Multiplatforms on AIX or Linux environments.

Initial installation
If you want to run an initial installation of System Automation for Multiplatforms, see “Running the installation” on page 20.

Existing installation

If a previous version of System Automation for Multiplatforms is already installed, you must run some steps before you can install the new version of System Automation for Multiplatforms. For more information about how to migrate a new version of the product, see “Migrating the system automation domain” on page 22.

Running the installation

Use an installation script to install System Automation for Multiplatforms.

The installation script runs the following actions:

- A complete prerequisite check to verify that all prerequisites are available and at the required level. If your system does not pass the check, the installation does not start, and you must provide the missing prerequisites before you restart the installation. Refer to “Checking prerequisites ” on page 3
- Install System Automation for Multiplatforms, including the end-to-end automation adapter.

To avoid having to restart the installation, you can start the prerequisites check separately before you begin the installation.

If an IBM Reliable Scalable Cluster Technology (RSCT) peer domain exists, ensure that the node on which you are running the script is offline in the domain. Otherwise, the installation is canceled.

Install the product, including the automation adapter:

1. Log in as root, or with equivalent authority.
2. If you downloaded the `.tar` file from the Internet, extract it:

   ```
   tar -xvf <tar file>
   ```

   If you received the product on a DVD, mount the DVD and change to the directory where the DVD is mounted.
3. Enter the following command:
   - Linux: `cd SAM4100MPLinux`
   - AIX: `cd SAM4100MPAIX`
4. Run the installation script:

   ```
   ./installSAM
   ```

   Typically, you do not need to specify any of the options that are available for the **`installSAM`** command. The default installation installs the packages for all supported languages. If you do not want to install all languages and want only the English language, you can specify the `--nonls` option. For a detailed description of the **`installSAM`** command, see *Tivoli System Automation for Multiplatforms Reference Guide* .
5. Read the information in the license agreement and the license information that is displayed. You can scroll forward line by line with the enter key, and page by page with the space bar, which is similar to the "more" option in UNIX. When you scrolled to the bottom of the license information file and you want to accept the terms of the license agreement, type 'y'. Any other input cancels the installation.

   The installation is also canceled if no license file is found.
6. After you accept the license agreement, the installation program checks prerequisites to verify that they are available and at the required level.

   If your system does not pass the check, the installation does not start, and you must provide the missing prerequisites before you restart the installation.

   Information about the results of the prerequisites check is available in the log file `/tmp/installSAM.<#>.log`.

   If your system passed the check, the product, including the automation adapter, is installed.

7. Check the following log file for information about the installation:

   ```
   /tmp/installSAM.<#>.log
   ```

   The hash symbol <#> is a number; the highest number identifies the most recent log file.

   The entries in the log file have the following prefixes:

   prereqSAM
   Entries that were written during the prerequisites check.

   installSAM
   Entries that were written during the installation of the product.
8. To find out which packages were installed, inspect `/tmp/installSAM.<#>.log`, where <#> is the highest number in the list of logs you find.

Installing the product license

System Automation for Multiplatforms requires that a valid product license is installed on each system it is running on.

The license is contained on the installation medium in the 'license' sub directory. The installation of the license runs during the product installation process. If the license was not installed successfully, run the following command to install the license:

```
samlicm –i license_file
```

To display the license, issue:

```
samlicm -s
```

For a detailed description of the command, see *Tivoli System Automation for Multiplatforms Reference Guide* .

Supported languages and locales

If you want to use System Automation for Multiplatforms in a language other than English, find out which languages and locales are supported.

Linux

Table 16 on page 21 shows the combinations of languages and locales that are supported for System Automation for Multiplatforms on Linux systems to display translated messages. New versions of Linux operating systems might not support all of the listed encoding. UTF-8 encoding is always supported.

Table 16. Languages and locales supported by System Automation for Multiplatforms on Linux systems.

Language	UTF-8	ISO-8859-1	EUC/GBK	Euro	GB18030/BIG5
German	de_DE.UTF-8	de_DE, de_DE.ISO-8859-1		de_DE@euro	
Spanish	es_ES.UTF-8	es_ES, es_ES.ISO-8859-1		es_ES@euro	
French	fr_FR.UTF-8	fr_FR, fr_FR.ISO-8859-1		fr_FR@euro	
Italian	it_IT.UTF-8	it_IT, it_IT.ISO-8859-1		it_IT@euro	

Table 16. Languages and locales supported by System Automation for Multiplatforms on Linux systems. (continued)

Language	UTF-8	ISO-8859-1	EUC/GBK	Euro	GB18030/BIG5
Japanese	ja_JP.UTF-8		ja_JP.eucJP		
Korean	ko_KR.UTF-8		ko_KR.eucKR		
Brazilian Portuguese	pt_BR.UTF-8	pt_BR			
Simplified Chinese	zh_CN.UTF-8		zh_CN.GBK, zh_CN.GB2312		zh_CN.GB18030
Traditional Chinese	zh_TW.UTF-8				zh_TW.Big5, zh_TW

AIX

The following table shows the combinations of languages and locales that are supported for System Automation for Multiplatforms on AIX to display translated messages.

Table 17. Languages and locales supported by Tivoli System Automation on AIX systems

Language	UTF-8	ISO-8859-1	EUC/GBK	SJIS/GB18030/ BIG5
German	DE_DE	de_DE		
Spanish	ES_ES	es_ES		
French	FR_FR	fr_FR		
Italian	IT_IT	it_IT		
Japanese	JA_JP		ja_JP	Ja_JP
Korean	KO_KR		ko_KR	
Brazilian Portuguese	PT_BR	pt_BR		
Simplified Chinese	ZH_CN		zh_CN	Zh_CN
Traditional Chinese	ZH_TW		zh_TW	Zh_TW

Migrating the system automation domain

You can migrate to System Automation for Multiplatforms version 4.1 if an older version of is already installed.

Before you can migrate one or more nodes to a newer level, make sure that you are familiar with the following characteristics:

- The migration process starts when any node within the active cluster is upgraded to the higher code level.
- You can always upgrade to a higher code level. Downward migration is not possible.
- The migration process is complete only when the active version number is equal to the highest installed code version number. Until then, different code levels can coexist.
- Starting with version 4.1, making the end-to-end automation adapter highly available does not require an automation policy any more. For more information, see “Migrating a highly available end-to-end automation adapter” on page 25.

Migrating an entire domain

During the migration, the domain is not available. To minimize downtime, you can run a prerequisites check before you start the actual migration.

For more information, see “Checking prerequisites ” on page 3.

Migrate an entire domain:

1. Make sure that all resources are offline:
   a. Check whether the end-to-end automation adapter is running:

      ```
      samadapter status
      ```

      If it is running, stop the automation adapter:

      ```
      samadapter stop
      ```

   b. Stop all online resource groups by setting their NominalState to Offline:

      ```
      chrg -o Offline <resource-group-name>
      ```

2. If the domain is online, stop the domain:

   ```
   stoprpdomain <domain-name>
   ```

3. On AIX, enter the following command after the cluster is stopped and before the installation is started:

   ```
   # /usr/sbin/slibclean
   ```

4. Run the `./installSAM` script from the installation directory on the product DVD or from the extracted electronic deliverable on all nodes. For more information about the `installSAM` script, see “Running the installation” on page 20.
5. Start the domain:

   ```
   startrpdomain <domain-name>
   ```

6. Check the code levels with the `lssrc –ls IBM.RecoveryRM` command (see the sample in “Verifying the active and installed version number” on page 24). All nodes have the newly installed code level, but the active code level is the previous one.
7. To activate the new version, continue with “Completing the migration” on page 24.

Migrating node-by-node

Node-by-node migration is supported only when you migrate from System Automation for Multiplatforms V2.3 or higher. To migrate the nodes of a domain one by one has the advantage that System Automation for Multiplatforms remains available during the migration.

For more information about how to minimize downtime, see “Checking prerequisites ” on page 3.

Run a node-by-node migration:

1. Exclude the node from automation to ensure that resources that must be kept available are moved to another node in the peer domain:

   ```
   samctrl -u a <node>
   ```

   Note: The command can run for a considerable amount of time until all move operations are complete.
2. Stop the node from another node in the domain, and verify that it is stopped:

   ```
   stoprpnode <node>; lsrpnode
   ```

3. To upgrade the node, run the script `./installSAM` from the installation directory on the product CD or from the extracted electronic deliverable. For more information about the `installSAM` script, see “Running the installation” on page 20.
4. Start the node:

```
startrpnode <node>
```

5. Include the upgraded node in automation again:

```
samctrl -u d <node>
```

6. The upgraded node can now join the existing domain. Use the `lssrc –ls IBM.RecoveryRM` command (see the sample in “Verifying the active and installed version number” on page 24) to display the installed version and the active version of the product. The new code features are not activated until the active System Automation for Multiplatforms version number is equal to the highest System Automation for Multiplatforms version number that is installed within the cluster. You cannot fully use these new code features until all the nodes are upgraded.
7. Repeat the steps 1-6 for other nodes within the cluster.
8. To activate the new version, continue with “Completing the migration” on page 24.

Verifying the active and installed version number

After the upgrade, the new features are not yet activated. The previous and new code levels can coexist until the migration is complete.

The **`lssrc –ls IBM.RecoveryRM`** command shows you the active version number AVN and the installed version number IVN of the product. When IVN and AVN are the same, migration is complete. Output:

```
Subsystem         : IBM.RecoveryRM
PID               : 31163
Cluster Name      : xdr43
Node Number       : 1
Daemon start time : 02/19/13 15:12:00

Daemon State:
   My Node Name         : lnxxdr43
   Master Node Name     : lnxxdr43 (node number = 1)
   Our IVN              : 4.1.0.0
   Our AVN              : 4.1.0.0
   Our CVN              : d4b7e876c (4b7e876c)
   Total Node Count     : 2
   Joined Member Count  : 2
   Config Quorum Count  : 2
   Startup Quorum Count : 1
   Operational Quorum State: HAS_QUORUM
   In Config Quorum     : TRUE
   In Config State      : TRUE
   Replace Config State : FALSE
```

Figure 9. Verifying the active and installed version numbers

To activate the new version, continue with “Completing the migration” on page 24.

Completing the migration

Check if the migration run successfully.

Check and complete the migration:

1. Make sure that the domain is started and that all nodes in the domain are online.
2. Issue the **`lsrpdomain`** command to display the version of RSCT that is active in the peer domain, and the mixed version status:

```
Name       OpState  RSCTActiveVersion  MixedVersions  TSPort  GSPort
SA_Domain  Online   2.5.5.1            Yes            12347   12348
```

3. Issue the **lsrpnode** command to display which version of RSCT that is installed on the nodes. Keep in mind that all nodes must be online:

```
Name   OpState RSCTVersion
node01 Online  2.5.5.1
node02 Online  2.5.5.1
node03 Online  2.5.5.1
```

4. If the RSCT peer domain is running in mixed version mode (`MixedVersions = Yes`) and all nodes are upgraded to the new release, update the active RSCT version by running the RSCT `CompleteMigration` action on one of the nodes. Before you can run the action, review the RSCT migration preparation procedures in *IBM RSCT Administration Guide*.

   To update the RSCT`ActiveVersion`, make sure that all nodes are online. Enter the following command on one of the nodes:

```
runact -c IBM.PeerDomain CompleteMigration Options=0
```

   To verify that the active RSCT version is updated, enter the **lsrpdomain** command again:

```
Name        OpState  RSCTActiveVersion  MixedVersions  TSPort GSPort
SA_Domain   Online   2.5.5.1            No             12347  12348
```

5. Run the **samctrl -m** command to activate the new features and to complete the migration. For more information about the command, see *System Automation for Multiplatforms Reference Guide*.
6. If the migration was done from System Automation for Multiplatforms release 3.1, you must adjust the value of the attribute `OperationalFlags` by entering the following command on one of the nodes:

```
chrsrc -c IBM.CHARMControl OperationalFlags=8088
```

   To display the actual value of this attribute, enter:

```
lsrsrc -c IBM.CHARMControl
```

The new code features are active if the value of the `ActiveVersion` and the `InstalledVersion` of System Automation for Multiplatforms is the same for all nodes.

Migrating a highly available end-to-end automation adapter

Find out how to upgrade a highly available end-to-end automation adapter to version 4.1.

Starting with System Automation for Multiplatforms version 4.1, an automation policy is not required any more to make the end-to-end automation adapter highly available. On Windows, this implementation was already available before version 4.1 and is now available for all other operating systems.

System Automation for Multiplatforms version 3.2 or lower:

Figure 10 on page 26 shows the environment in which the end-to-end automation adapter operated in UNIX and Linux clusters.

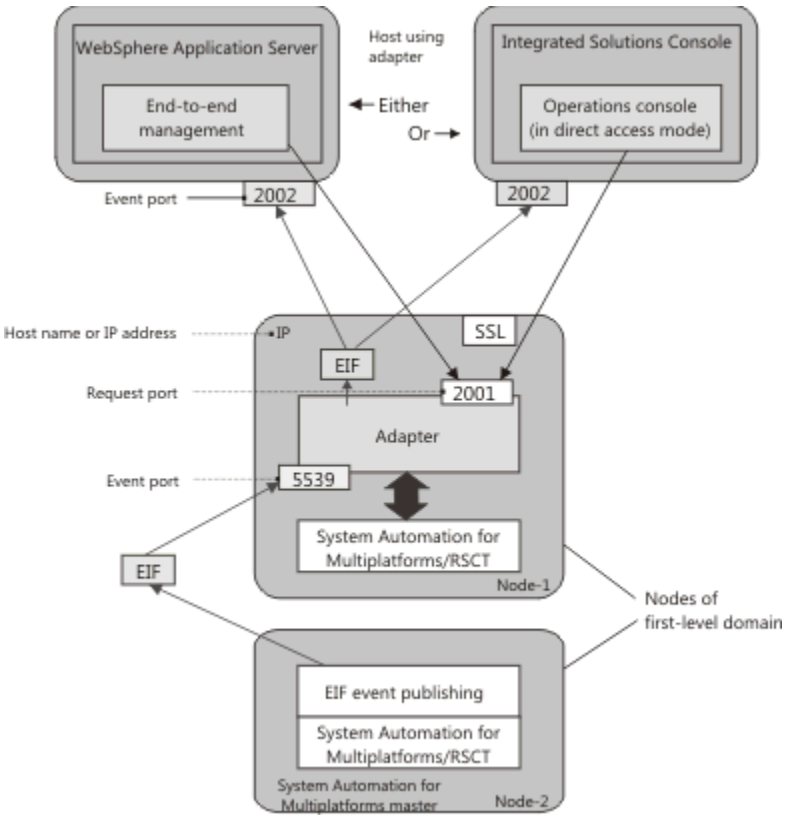

Figure 10. End-to-end automation adapter environment in UNIX and Linux clusters before version 4.1

System Automation for Multiplatforms version 4.1:

Figure 11 on page 26 shows the environment in which the adapter operates starting with version 4.1.

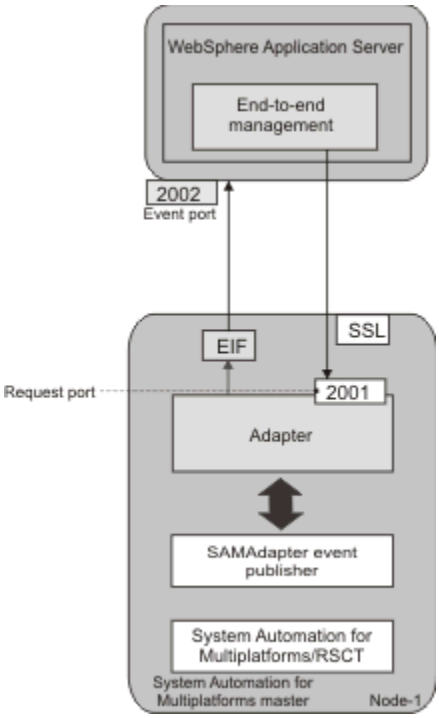

Figure 11. End-to-end automation adapter environment available with version 4.1

Starting with version 4.1, the automation adapter is attached to the System Automation master node. The cluster infrastructure makes sure that the System Automation for Multiplatforms master and the adapter are always available. No additional automation policy is required to make the adapter highly available. The virtual IP address, which is a System Automation critical resource is not further required in this scenario.

Changing the high availability implementation for the end-to-end automation adapter has the following implications if you upgrade your cluster node by node as described in "Migrating node-by-node" on page 23.

Old implementation is active during migration process
If the migration is not yet complete, there are different versions of the code active on different nodes in the cluster. During this time, the old high availability implementation is still active. The new implementation becomes active as soon as the active version is set to version 4.1.0.0 (or higher). For more information, see "Completing the migration" on page 24

Automation policy configuration is still possible during migration process
An automation policy is not required anymore to make the adapter highly available. Nevertheless, the old implementation is still supported if you did not complete a node-by-node migration. The adapter automation policy configuration tasks are still available and supported during the migration process. The documentation of those configuration tasks is removed. If you still need to inspect the description of these tasks, refer to the documentation for the previous version of the product.

Note: If you want to modify the high availability configuration during a node-by-node migration, make sure to run the configuration utility on a cluster node that is online. The reason is that the active version number cannot be determined on an offline node. The old implementation of the adapter high availability configuration is not available in an offline cluster or on an offline node. Even if the active version number is lower than 4.1.0.0.

Before you complete the migration to version 4.1, check all your automation policies for an entire domain and node-by-node migration. The automation policies can contain resources that are related to the end-to-end automation adapter high availability. Remove all those resources:

- Check which resource prefix you use when you configure the adapter automation. The default prefix is `samadapter-`.
- Remove all relationships, resources, and resource groups that have a name that is starting with the prefix.
- If you are using the `xml` format to define your policies, remove all relationships, resources, and resource groups that have a name that is starting with the prefix from the `xml` files.

Required actions after you completed the migration
If the adapter is running while the cluster migration is initiated, then the adapter will be stopped and not started again after the migration completed.

Perform the following manual migration steps before you can start the adapter:

1. Run the configuration utility `cfgsamadapter` to change the adapter host name or IP address. Select the local host name of each cluster node as default or specify a distinct host name or IP address.
2. If you select the default for the adapter host, replicate the configuration to the other nodes in the cluster. Otherwise, explicitly configure a host name or IP address on each cluster node.

You can now start the adapter. You can use the configuration dialog as described in System Automation for Multiplatforms Administrator's and User's Guide or use the `samadapter start` command.

Continue to use old adapter high availability implementation
In some rare cases, you might not be able to use the new adapter high availability implementation. For example, if you want to enforce that the adapter runs only on a subset of the available nodes in the cluster. This scenario is possible with the former automation policy. But with the new approach the adapter might run on any cluster node.

In such a case, you have the opportunity to enforce that the automation policy is still used to make the adapter highly available if you are using version 4.1. Even if you are already running a cluster by using

either the new or the old approach, you can switch to the respective other approach. The following scenarios are supported:

1. **Continue to use old implementation when you migrate to version 4.1.**

   If you migrate your cluster from a version lower than 4.1 to version 4.1, the new adapter high availability implementation is activated. If you want to use the old implementation instead, run the following steps after you upgraded the product code to version 4.1 on all cluster nodes:

   a. Edit the configuration properties file `/etc/opt/IBM/tsamp/sam/cfg/sam.adapter.properties` and change the value of parameter `use-adapter-ha-policy` from `false` to `true` on each node in the cluster.

   b. Issue the command **`samctrl -m`**.

2. **Switching to the new adapter high availability implementation after you completed the migration.**

   If you migrated your cluster from a version lower than 4.1 to version 4.1 and you followed the procedure that is described for scenario 1 above, you are still using the old adapter high availability implementation. If you then want to switch to the new adapter high availability implementation, run the following steps:

   a. Stop the domain by entering the command **`stoprpdomain`**.

   b. Edit the configuration properties file `/etc/opt/IBM/tsamp/sam/cfg/sam.adapter.properties` and change the value of parameter `use-adapter-ha-policy` from `true` to `false` on each node in the cluster.

   c. Start the domain by entering the command **`startrpdomain`**.

3. **Switching back to the old adapter high availability implementation after you completed the migration.**

   If you complete the migration of your cluster from a version lower than 4.1 to version 4.1 without following the procedure that is described for scenario 1 above, you are using the new adapter high availability implementation. The same is true if you followed the procedure that is described for scenario 2. If you then want to switch back to the old adapter high availability implementation, run the following steps:

   a. Stop the adapter by entering the command **`samadapter stop`**.

   b. Edit the configuration properties file `/etc/opt/IBM/tsamp/sam/cfg/sam.adapter.properties` and change the value of parameter `use-adapter-ha-policy` from `false` to `true` on each node in the cluster.

   c. Start the configuration utility by entering the command **`cfgsamadapter`** and complete the following tasks:

      i) On the main window of the configuration dialog, click **Configure**.

      ii) Click **Save** to save the configuration changes. Then, the EEZ publisher entry that is required for the old implementation is added to the configuration properties file `/etc/Tivoli/tec/samPublisher.conf` in any case. This is required because the publisher entry can be removed by the adapter when it uses the new adapter high availability implementation.

      iii) On the main window of the configuration dialog, click **Replicate** and propagate the configuration changes to the other nodes in the cluster.

      iv) On the main window of the configuration dialog, click **Define** to activate the adapter high availability policy again. It is removed by System Automation for Multiplatforms during execution of the **`samctrl -m`** command.

   d. Start the adapter by entering the command **`samadapter start`**.

4. **Using the old adapter high availability implementation in a new version 4.1.0.0 cluster.**

   If you run an initial installation of version 4.1.0.0, you are using the new adapter high availability implementation. If you want to use the old adapter high availability implementation instead, run the following steps:

a. Stop the adapter by entering the command **samadapter stop**.

b. Edit the configuration properties file `/etc/opt/IBM/tsamp/sam/cfg/sam.adapter.properties` and change the value of parameter `use-adapter-ha-policy` from `false` to `true` on each node in the cluster.

c. Start the configuration utility by entering the command **cfgsamadapter** and complete the following tasks:

   i) On the main window of the configuration dialog, click **Configure**.

   ii) On the **Automation** tab, configure the adapter high availability policy.

   iii) Click **Save** to save the configuration changes.

   iv) On the main window of the configuration dialog, click **Replicate** and propagate the configuration changes to the other nodes in the cluster.

   v) On the main window of the configuration dialog, click **Define** to activate the adapter high availability policy.

d. Start the adapter by entering the command **samadapter start**.

Scenarios 3 and 4 refer to the **Define** task and the **Automation** tab. The corresponding documentation is removed in version 4.1. If you still need to inspect the description of these tasks, refer to the documentation for the previous version of the product.

Postinstallation

To be able to obtain the debug data, you must configure the system logger.

After you installed System Automation for Multiplatforms on AIX, you must complete the following task:

Configure the system logger on AIX

The system logger is not configured by default. Messages are written to the error log.

To be able to obtain the debug data, you must configure the system logger in the file `/etc/syslog.conf`. When you made the necessary changes, you must recycle the `syslogd` with the **refresh –s syslogd** command. The location of the log file is defined in `/etc/syslog.conf`.

No further action is required in the case of Linux.

Making shared volume groups enhanced concurrent capable on AIX

If your shared volume groups are not enhanced concurrent capable, when a node crashes, the disks are locked and the remote node is not able to access the disk. To avoid this situation, make the shared volume group enhanced concurrent capable.

Note:

1. Ensure that the package `bos.clvm.enh` is installed on your system.
2. System Automation for Multiplatforms supports enhanced concurrent capable volume groups in non-concurrent mode when using resources of classes `IBM.AgFileSystem` or `IBM.VolumeGroup` within the policy. System Automation for Multiplatforms does not support the concurrent mode of enhanced concurrent volume groups and the contained filesystem as resources within the policy. Support for enhanced concurrent volume groups can be explicitly provided by a policy provider using `IBM.Application` resources to manage the filesystem on top of the enhanced concurrent volume groups.

Before you make the volume group enhanced concurrent capable, use the `lsvg` command to view information about the shared volume group:

```
# lsvg vgERSTZ0
VOLUME GROUP:       vgERSTZ0                 VG IDENTIFIER:
00c31bfe00004c0000000118c2f1ead2
VG STATE:           active                   PP SIZE:        4 megabyte(s)
VG PERMISSION:      read/write               TOTAL PPs:      255 (1020
megabytes)
MAX LVs:            256                      FREE PPs:       14 (56 megabytes)
```

```
LVs:                2                       USED PPs:       241 (964 megabytes)
OPEN LVs:           2                       QUORUM:         2 (Enabled)
TOTAL PVs:          1                       VG DESCRIPTORS: 2
STALE PVs:          0                       STALE PPs:      0
ACTIVE PVs:         1                       AUTO ON:        no
MAX PPs per VG:     32512
MAX PPs per PV:     1016                    MAX PVs:        32
LTG size (Dynamic): 256 kilobyte(s)         AUTO SYNC:      no
HOT SPARE:          no                      BB POLICY:      relocatable
```

To make a volume group enhanced concurrent capable using SMIT:

1. Enter the following command:

   ```
   # smitty vg
   ```

   Text similar to the following is displayed:

   ```
         Set Characteristics of a Volume Group
             Change a Volume Group

   * VOLUME GROUP name                                   vgERSTZ0
   * Activate volume group AUTOMATICALLY                 no                      +
      at system restart?
   * A QUORUM of disks required to keep the volume       yes                     +
      group on-line ?
     Convert this VG to Concurrent Capable?             enhanced concurrent
   ```

2. Press ENTER.

To make the volume group enhanced concurrent capable from the command line, enter:

```
# /usr/sbin/chvg -a'n' -Q'y' '-C' <VOLUME_GROUP_NAME>
```

After you made the volume group enhanced concurrent capable, the `lsvg` command returns information similar to the following example output:

```
# lsvg vgERSTZ0
VOLUME GROUP:       vgERSTZ0                VG IDENTIFIER:
00c31bfe00004c0000000118c2f1ead2
VG STATE:           active                  PP SIZE:        4 megabyte(s)
VG PERMISSION:      read/write              TOTAL PPs:      255 (1020
megabytes)
MAX LVs:            256                     FREE PPs:       14 (56 megabytes)
LVs:                2                       USED PPs:       241 (964 megabytes)
OPEN LVs:           2                       QUORUM:         2 (Enabled)
TOTAL PVs:          1                       VG DESCRIPTORS: 2
STALE PVs:          0                       STALE PPs:      0
ACTIVE PVs:         1                       AUTO ON:        no
Concurrent:         Enhanced-Capable        Auto-Concurrent: Disabled
VG Mode:            Non-Concurrent
MAX PPs per VG:     32512
MAX PPs per PV:     1016                    MAX PVs:        32
LTG size (Dynamic): 256 kilobyte(s)         AUTO SYNC:      no
HOT SPARE:          no                      BB POLICY:      relocatable
```

Rollback procedure

Follow the steps described to roll back your installation to the previous release.

To roll back System Automation for Multiplatforms to the previous release, run the following steps:

1. Save the automation policy:

   ```
   sampolicy -s file.xml
   ```

2. Change all resource groups to offline. Alternatively, if you do not want to impact the resources, stop the domain:

   ```
   stoprpdomain –f domain_name
   ```

3. Remove the domain if you also need to roll back the RSCT level. Otherwise, bring the domain offline.

4. System Automation for Multiplatforms can be rolled back by running the command `./installSAM --forceAll`. The command installs the System Automation for Multiplatforms and RSCT deliverable where `installSAM` is located regardless of the version, which is already installed.
5. If you removed the domain, create the domain again. Otherwise, start the domain by entering `startrpdomain -w domain_name`.
6. If the domain was created again, you can reapply the saved policy by entering `sampolicy -a file.xml`.

Uninstalling

You can remove System Automation for Multiplatforms from your AIX and Linux environments with a documented procedure.

Consider the following hints before you start the deinstallation procedure:

- Use the **uninstallSAM** script that is provided for your operating system to uninstall System Automation for Multiplatforms. For example, run **./uninstallSAM** from the installation directory, to ensure a correct deinstallation of the product.
- Before uninstalling, always save your configuration with the **sampolicy -s** command. For more information about how to save a System Automation for Multiplatforms configuration, see *Tivoli System Automation for Multiplatforms Administrator's and User's Guide*.

  The description of the **sampolicy** command in *System Automation for Multiplatforms Reference Guide*.
- The command **uninstallSAM** removes all configuration information that you defined for the domain. For this reason, you must never use **uninstallSAM** if you intend to upgrade to a new version.

To uninstall System Automation for Multiplatforms, run the following steps:

1. Ensure that the domain is offline:
   - To check whether a domain is online, run the following command:

     ```
     lsrpdomain
     ```

   - To stop the domain, run the following command:

     ```
     stoprpdomain <domain>
     ```

2. Uninstall the product with the `uninstallSAM` script available in the `/opt/IBM/tsamp/sam/uninst/` directory:

   ```
   ./uninstallSAM
   ```

   Typically, you do not need to specify any of the options that are available for the `uninstallSAM` command. For a detailed description of the command, see System Automation for Multiplatforms Reference Guide.

   Redhat Package Manager ensures that RSCT and SRC are not uninstalled with System Automation for Multiplatforms, in case CSM or GPFS is installed on the same Linux system. CFM or GPFS also use the RSCT and System Resource Controller (SRC) packages. Redhat Package Manager messages indicate this condition.
3. Check the following log file for information about the deinstallation:

   ```
   /tmp/uninstallSAM.<#>.log
   ```

   The hash symbol <#> indicates a number. The highest number identifies the most recent log file.
4. To verify which packages were uninstalled, inspect `/tmp/uninstallSAM.<#>.log`, where <#> is the highest number of the log files, which you can find.

Note: The command `uninstallSAM` deletes all settings that are stored under `/etc/opt/IBM/tsamp/sam` as well.

Installing on new operating systems

New operating system support can be introduced with a fix pack 4.1.0.<f>, where <f> is the respective fix pack number

An installation of System Automation for Multiplatforms 4.1.0.0 as described in “Installing System Automation for Multiplatforms” on page 19 is only possible on the set of platforms and operating system versions that are initially supported for this version 4.1 release level. Nevertheless, support for additional platforms or operating system versions can be added with fix packs at a later point in time. This is referred to as "new platform support" in the following. If you want to install a fix pack on an already supported operating system, upgrade you installation.

To check which new platform support is introduced with which fix pack, see “Supported platforms” on page 4.

If new operating system support is introduced with a fix pack, this fix pack must be installed as an initial installation rather than an upgrade installation. Therefore, it is required to copy the 4.1 license file into the `SAM410<f>MP<platform>/license` directory before you start the fix pack installation. Perform the following steps:

1. Obtain a System Automation for Multiplatforms license file that is contained in one of the 4.1 release deliverables:

   Product DVD
   Use one of the DVDs listed in “Product DVD” on page 1 to obtain the license. You find the license file that is named `sam41.lic` in directory `SAM4100MP<platform>/license`.

   Electronic distribution
   Use one of the archive files that are listed in “Electronic distribution” on page 1 to obtain the license. Extract the archive file. In the expanded directory tree, you find the license file that is named `sam41.lic` in the directory `SAM4100MP<platform>/license`.
2. Extract the 4.1.0.<f> fix pack archive file that contains the new operating systems support as described in “Usage instructions for platform-specific archives” on page 33. In the expanded directory tree, the directory `SAM410<f>MP<platform>/license` is empty.
3. Copy the license file that is obtained in step 1 into the `SAM410<f>MP<platform>/license` directory of the expanded fix pack directory tree.
4. Start the System Automation for Multiplatforms installation as described in “Running the installation” on page 20. The installation program runs an initial installation of the product on the new operating system.

Migration from SLES 12 to SLES 15, or from RHEL 8 to RHEL 9

You can migrate from SLES 12 to SLES 15 or from RHEL 8 to RHEL 9 with the existing System Automation for Multiplatforms clusters.

Run the following steps to migrate your cluster:

1. Save the policy by using the `sampolicy -s` command. Stop the resources and remove the domain.
2. Install the target OS platform SLES 15 or RHEL 9 on all the cluster nodes.
3. Install the System Automation for Multiplatforms package supporting SLES 15 and RHEL 9: 4.1.1-TIV-SAMP-Linux64-FP000x. For more information, see Installing on new operating systems.
4. Create the domain again, and activate the policy: `sampolicy -a`

Note:

1. The Node-by-Node migration feature is only supported to upgrade the System Automation for Multiplatforms product level, but not to upgrade the operating system version.
2. It is not supported to have a domain with mixed operating system levels, for example: SLES 12/15 RHEL 8/9.

Installing service fix packs

Installing service means applying corrective service fix packs to release 4.1 of System Automation for Multiplatforms or upgrading the software release level from release 4.1. Such service fix packs are referred to as product fix packs.

Product fix packs are available for System Automation for Multiplatforms in the following formats:

Linux
Archives in compressed `.tar` format.

AIX
Archives in compressed `.tar` format.

Obtaining fix packs

For more information, refer to the System Automation for Multiplatforms Product page.

Archives for product fix packs can be downloaded from the System Automation for Multiplatforms Support Portal . Download the archive to a temporary directory. Typically, one archive is available for each operating system. For more information about the naming conventions that apply to product fix pack archives, see “Archive naming conventions” on page 33.

Archive naming conventions

Learn more about the syntax of the archive names.

The archives for product fix packs for the System Automation for Multiplatforms have the following syntax:

`4.1.0-TIV-SAMP-<platform>-FP<fix_pack_number>.<archive_type>` contains the service fix pack for System Automation for Multiplatforms.

Explanation:

`<platform>`
Operating system on which System Automation for Multiplatforms is installed.

`<fix_pack_number>`
Fix pack number.

`<archive_type>`
Either `tar.gz` or `tar.Z`.

Example:

The `tar.Z` archive that is used to install fix pack 1 for System Automation for Multiplatforms 4.1.0 on AIX operating systems:

```
4.1.0-TIV-SAMP-AIX-FP0001.tar.Z
```

Usage instructions for platform-specific archives

Learn more about how to download and install the fix pack.

The tables list the archive files, which you can download for applying service for the Linux and AIX operating systems. For each archive, follow the specific instructions that are listed in the **Description** column.

Linux
The following table describes the System Automation for Multiplatforms archive file that contains the corresponding 64-bit service deliverable. For more information, see “Installing on new operating systems” on page 32.

Table 18. Archive for Linux 64-bit operating systems	
Archive name	**Description**
`4.1.0-TIV-SAMP-Linux64-FP<fix_pack_number>.tar.gz`	Use the **tar -zxf** command to decompress and extract the archive. After you extracted the archive, the installation script `installSAM` is stored in: `SAM41<maintenance_level>MPLinux64/installSAM`

AIX

Table 19. Archive for AIX operating systems	
Archive name	**Description**
`4.1.0-TIV-SAMP-AIX-FP<fix_pack_number>.tar.Z`	Use the **uncompress** command to decompress the archive, then use the `tar -xf` command to extract the archive. You can find the installation script `installSAM` after you extracted the archive: `SAM41<maintenance_level>MPAIX/installSAM`

Installing service for System Automation for Multiplatforms

Installing service means upgrading System Automation for Multiplatforms from release 4.1. Therefore, release 4.1 must be installed before any service can be applied.

Before you begin:

- Product fix packs are always cumulative.
- You must have root authority to install a product fix pack.
- When you downloaded the archives from the System Automation for Multiplatforms support site (see "Obtaining fix packs" on page 33), unpack the product fix pack archive to a temporary directory. For information about how to unpack the archive for your operating system, see "Usage instructions for platform-specific archives" on page 33.
- Back up your system configuration before you install the service fix pack. For more information, see *Tivoli System Automation for Multiplatforms Administrator's and User's Guide*.
- To minimize downtime, you can run a prerequisites check before you start the installation. For more information, see "Checking prerequisites " on page 3.

Perform the following steps on each node in the peer domain:

1. Check whether any resources are online on the node you want to service:
   - If resources are online and must be kept available, exclude the node from automation:

     ```
     samctrl -u a <node>
     ```

     System Automation for Multiplatforms stops the resources on the node and, if possible, restarts them on a different node in the peer domain.
   - If the resources need not to be kept available during service, bring the resource groups offline.
2. Stop the node from another node in the domain, and verify that it is stopped:

   ```
   stoprpnode <node>; lsrpnode
   ```

3. After you received the archives, extract them. They create a directory structure with root directory SAM41`mf`MP, where `mf` stands for modification level and fix level.
4. Install the service fix pack with the `installSAM` script. For detailed information about the script, refer to "Running the installation" on page 20.
5. Start the node:

```
startrpnode <node>
```

6. If you excluded the node in step 2, include the node to the automation:

```
samctrl -u d <node>
```

7. If you require the resource groups to be online, bring the resource groups online. Otherwise, delay this step until the last node in the peer domain is serviced.
8. After all nodes are serviced, run the steps that are described in "Completing the migration" on page 24. The changes become effective in the entire domain and the correct version is shown.

Uninstalling service

To uninstall a fix pack, you need to uninstall the complete product.

To uninstall System Automation for Multiplatforms, follow the instructions in "Uninstalling" on page 31.

After the uninstallation is complete, you can reinstall System Automation for Multiplatforms and the required service level (fix pack level).

Installing the extended disaster recovery (xDR) feature

Today, businesses and companies depend on disaster recovery solutions to recover critical data. To solve this issue, System Automation for Multiplatforms supports GDPS/PPRC Multiplatform Resiliency on System z (xDR).

Geographically Dispersed Parallel Sysplex® (GDPS®) is an application availability and disaster recovery solution, which is highly customized to work with your z/OS® environment. It provides disaster and failure recovery from a single point of control and ensures data consistency. For more information about GDPS, see the IBM Redbooks® publication *GDPS Family - An Introduction to Concepts and Capabilities*, which can be downloaded at IBM Redbooks.

System Automation for Multiplatforms extends GDPS/PPRC for Linux systems that are running on System z. It provides a coordinated disaster recovery solution for systems that are running on

- zSeries, including z/OS
- Linux on System z under z/VM
- Linux on System z running native in the LPAR

xDR packaging

The code of the xDR feature is included as part of the System Automation for Multiplatforms product. You cannot use the corresponding function unless you installed a separate license to enable the code.

You can get the license when you order the xDR feature. The name of the license file is `sam41XDR.lic`:

DVD
Install the xDR feature from the DVD System Automation for Multiplatforms v4.1 – xDR for Linux on System z. The license file is available in the directory `SAM4100FeatXDR/license`.

Electronic distribution
If you obtain the xDR feature through electronic distribution, you find the license file in the electronic distribution file `CIVG7ML.txt`. This file is identical to the license file itself. Rename or copy the electronic distribution file to `sam41XDR.lic`.

xDR prerequisites

Before you can install the xDR feature license, you must install the System Automation for Multiplatforms base product.

xDR is only supported on Linux on System z.

The following Linux distributions are supported for xDR:

- xDR for Linux on System z running under z/VM® requires one of the following operating systems:
  - SUSE SLES 12 (64-bit)
  - SUSE SLES 15 (64-bit)
  - Red Hat RHEL 7 (64 bit)
  - Red Hat RHEL 8 (64 bit)
- xDR for Linux on System z running native in LPAR by using ECKD™ disks requires one of the following operating systems:
  - SUSE SLES 12
  - SUSE SLES 15

Note:

1. If you want to use the xDR function, particular versions of z/VM, Linux on System z, GDPS, and System Automation for Multiplatforms must be installed. For detailed information about the available function and the required versions, refer to the GDPS manuals. System Automation for Multiplatforms supports xDR for Linux on System z only.
2. The xDR naming conventions require that the names of clusters and nodes must not exceed 32 characters. Cluster and node names are not allowed to contain periods (.) or dashes (-) and they must not be identical. For xDR, cluster names are not case-sensitive. To use xDR, System Automation for Multiplatforms must be customized as described in the GDPS manuals.
3. English is the only language that is supported by xDR and GDPS.

Installing the xDR feature license

Use the **samlicm** command to install the license.

The license file must be accessible from the system where System Automation for Multiplatforms is installed. Copy file `sam41XDR.lic` to a location where it is accessible when you start **samlicm**.

Install the license:

```
samlicm -i  <license file location>/sam41XDR.lic
```

Verify that the feature license is successfully installed:

```
samlicm -s
```

The name of the xDR feature appears as value of the `Product Annotation` field in the output of the command. For example:

```
...
Product ID: 101
Product Annotation: SA for MP xDR for Linux on System z
...
```

For more information about the **samlicm** command, see *System Automation for Multiplatforms Reference Guide*.

Upgrading the xDR feature from a version lower than 4.1

Starting with version 4.1, the xDR feature license is installed into a different target directory.

If you upgrade the xDR feature from a version lower than 4.1, the previous installed xDR feature license is removed. Install again the feature license as described in “Installing the xDR feature license” on page 36. You can use the license file of the System Automation for Multiplatforms version from which you upgraded the product code. Or you can use the license file of the version to which you upgraded to.

Starting with version 4.1 xDR for Linux on System z® running under z/VM® supports only that the storage for all proxy nodes is permanently locked. Customers that are currently using a dual node proxy cluster with the option to lock the storage for the master proxy must migrate by running the script `enableErpd`.

Locking of the storage must then be done by adding the LOCK command to the `boot.local` or `rc.local` file of both proxy nodes. For more information, see the GDPS manuals.

Uninstalling the xDR feature

There is no specific uninstallation procedure that is defined for the xDR feature. It is uninstalled implicitly when System Automation for Multiplatforms is uninstalled.

Installing the SAP high availability policy

The SAP Central Services high availability policy feature is included as part of System Automation for Multiplatforms but requires a separate license.

The SAP high availability policy feature is included as part of System Automation for Multiplatforms but requires a separate license.

For more information about how to install the SAP high availability policy feature, see System Automation for Multiplatforms High Availability Policies Guide.

Chapter 3. Configuring

After you successfully installed System Automation for Multiplatforms, process configuration tasks that depend on the System Automation for Multiplatforms components and functions that you require.

Note: You need an X11 server to use the automation adapter configuration dialog. You need the 32-bit version of the X11 installation packages to run the configuration dialog. On some Linux operating systems, those packages are contained on the distribution media, but are not part of the standard installation. Make sure that the 32-bit version of the X11 installation packages is installed.

You can also configure the automation adapter in silent mode by using an input properties file. If an X11 server is not available, silent configuration is the only supported method on this system. For more information, see "Configuring in silent mode" on page 74.

Configuring the system automation behavior

You can manage and control System Automation for Multiplatforms by changing a set of attributes that affect product behavior.

You can start or stop the automation function, define timeout periods, and exclude nodes from automation, for example, for maintenance purposes.

You can modify the following attributes:

TimeOut
Specifies the timeout value in seconds for a start control operation that is run by System Automation for Multiplatforms. When the timeout period expires, the operation is repeated if the RetryCount is not exceeded.

RetryCount
Number of times a control operation can be tried again if it fails or times out.

Automation
Flag to enable or disable automation by System Automation for Multiplatforms.

ExcludedNodes
List of nodes on which System Automation for Multiplatforms actively pushes resources away or stops them. Can be used for maintenance purposes, for example.

ResourceRestartTimeOut
Amount of time in seconds System Automation for Multiplatforms waits to restart resources, which were on a failed node on another node.

TraceLevel
The trace level can be used to control the number of trace entries written. The maximum value of 255 results in detailed tracing, while the value 0 suppresses writing various classes of trace entries. Lowering the trace level is advisable for automation policies with many resources.

You can list the current values of the attributes with the command **`lssamctrl`**. The attributes are changed with the **`samctrl`** command. For more information, see *IBM Tivoli System Automation for Multiplatforms Reference* for a listing and description of these commands.

TimeOut and RetryCount

The `TimeOut` attribute is always used in conjunction with the `RetryCount` attribute:

TimeOut
Specifies how long System Automation for Multiplatforms will wait for a resource manager to do something.

RetryCount
Specifies the number of possible control operation attempts System Automation for Multiplatforms makes within the `TimeOut` period if the control operation is not successful. In general, if the first

attempt is not successful, the chances are fairly low that it works on the second or subsequent attempts.

Start operations

The operation timer is started when System Automation for Multiplatforms sends the first resource start control operation to a resource. When the timer has started, there are three possibilities:

1. The resource changes to the desired state (online or offline) within the timeout period. In this case, no further actions are triggered because the resource is in the state in which System Automation for Multiplatforms wants it to be.
2. The resource rejects the start control operation within the timeout period. What happens next depends on the reject code:
   - If it indicates that the error is recoverable, System Automation for Multiplatforms will continue to issue start control operations against the resource. Every control operation attempt is counted. When the RetryCount value is exceeded, System Automation for Multiplatforms stops issuing further control operations .
   - If the error is not recoverable, the resource will go into a problem state. Whether or not this triggers further automation actions depends on the type of resource against which the start operation was issued:
     – If a fixed resource is affected, no further actions are triggered.
     – If the control operation was issued against a constituent of a floating resource and this constituent is in state Offline or Failed Offline, System Automation for Multiplatforms will attempt to issue the control operations against another constituent of the resource. Note that the constituent that rejected the control operation will remain in an unrecoverable error state until you issue a reset operation against it.
3. The resource does not reach the desired state (online) within the timeout period. In this case, System Automation for Multiplatforms first issues a reset operation against the resource and waits until the reset operation has been accepted and the resource is offline. Then, System Automation for Multiplatforms issues another start control operation against the resource. Every control operation attempt is counted and System Automation for Multiplatforms stops issuing control operations when the RetryCount is exceeded or when the maximum timeout (TimeOut * RetryCount) expires, whichever comes first.

When System Automation for Multiplatforms stops issuing control operations for a fixed resource or for a constituent of a floating resource, the OpState of the resource is set to failed offline. This indicates that the resource is no longer usable and that manual intervention is required to correct the cause of the failure. When the problem has been resolved, the resource must be reset with the RMC command **`resetrsrc`**.

Note the retry counter is always reset when the resource reaches its desired state because no threshold is implemented. This means, for example, that a resource that is started, stays online for a short period of time, and then stops again, will be restarted by System Automation for Multiplatforms in a loop.

Default values are:

- TimeOut = 60
- RetryCount = 3

You use the command **`samctrl -t Timeout`** to change the TimeOut value and the command **`samctrl -r Retry_count`** to change the RetryCount value.

The `IBM.Application` class provides its own timeout value. If you add a resource of class `IBM.Application` to a group, the general `TimeOut` value is not used for this resource. As `TimeOut` value for this group member the larger value of `StartCommandTimeout` or `MonitorCommandPeriod` attribute (which are attributes of the `IBM.Application` resource) is used.

Stop operations

The operation timer is started when System Automation for Multiplatforms first sends a resource stop control operation to a resource. After the timer has started, there are three possibilities:

1. The resource changes to the desired state (offline) within the timeout period. No further actions are triggered.
2. The resource rejects the stop control within the timeout period. What happens next depends on the reject code:
   - If it indicates that the error is recoverable, System Automation for Multiplatforms issues another stop control operation against the resource.
   - If the error is not recoverable, the resource goes into a problem state. Manual intervention is required to get the resource out of the problem state.
3. The resource does not reach the desired state (offline) within the timeout period. In this case, System Automation for Multiplatforms first issues a reset operation against the resource and waits until the resource reaches its desired state (offline).

Automation

This flag indicates whether the System Automation for Multiplatforms automation function is enabled or not. If automation is disabled, System Automation for Multiplatforms stops sending control operations. The state of resources remain unchanged.

The default value is AUTO mode, which means that automation is turned on.

You use **samctrl –M F** to enable automation, **samctrl –M T** to disable automation.

ExcludedNodes

List of nodes on which System Automation for Multiplatforms stops all resources and moves them to another node if possible.

For example, you have floating resource A, which can run on four nodes node05, node06, node07, and node08. It is a member of resource group RG_A. After you made the group online, it is started on node05. If you add node05 to the list of excluded nodes, System Automation for Multiplatforms stops the resource on node05. The resource is restarted on one of the other nodes.

Caution: If you exclude a node and one or more mandatory members of a group cannot be restarted on another node, the whole group may be stopped.

By default, the list is empty, which means that all nodes in the peer domain can be used.

Use **samctrl -u a** to add one or more nodes to the list of excluded nodes. **samctrl -u d** to delete nodes from that list. **samctrl -u r** to replace nodes in the list.

ResourceRestartTimeout

The `ResourceRestartTimeout` value specifies the time in seconds that System Automation for Multiplatforms waits before it restarts resources that are on a failed and different node. Resources or the failed node can clean up before the resources are moved to another system.

The default value is 5 seconds.

You specify the resource restart timeout value with the command **samctrl –o**.

You can specify the trace level with the command **samctrl -l**. The TraceLevel determines the number of trace entries that are written. The default value is 127. The maximum value of 255 results in detailed tracing. If the value is set to 0, various classes of trace entries are not written. Reducing the trace level is advisable for automation policies with many resources.

Examples

To list the current System Automation for Multiplatforms control parameters you use the **lssamctrl** command.

System Automation for Multiplatforms control information:

```
SAMControl:
                TimeOut                             = 60
                RetryCount                          = 3
                Automation                          = Auto
                ExcludedNodes                       = {}
                ResourceRestartTimeOut      = 5
                ActiveVersion                       = [4.1.0.0,Thu Sept 27 11:10:58 METDST 2012]
                EnablePublisher                 = XDR_GDP2 XDR_GDP1
                TraceLevel                          = 31
                ActivePolicy                    = []
                CleanupList                         = {}
                PublisherList                       = {}
```

To add the node node05 to the list of excluded nodes, enter:

```
samctrl -u a node05
```

To set the `RetryCount` parameter to 5, enter:

```
samctrl –r 5
```

Configuring the tiebreaker

Configure a tiebreaker for cluster environments with an even number of nodes.

System Automation for Multiplatforms requires the majority of nodes to be online in the domain to start automation actions. If the domain consists of an even number of nodes, it might happen that exactly half of the nodes of the domain are Online. In this case, System Automation uses a tiebreaker to decide the quorum state, which determines whether automation actions can be started (**HAS_QUORUM**), or if no automation actions are possible (**PENDING_QUORUM, NO_QUORUM**).

Configure a shared disk tiebreaker such as ECKD or SCSI by using the **IBM.TieBreaker** resource class. Additionally two tie breakers are predefined, operator and fail. The operator tiebreaker provides an undetermined result when a tie occurs and it is left to the administrator to resolve the tie through granting or denying the operational quorum. When a tie occurs and a tiebreaker of type Fail is active, the attempt to reserve the tiebreaker is always denied. The default tie breaker type is set to Operator.

Additional implementations of a tiebreaker can be added by using the tiebreaker type **EXEC**. System Automation for Multiplatforms provides a network and an NFS tiebreaker as additional tiebreaker implementations.

List the available tiebreaker type:

```
lsrsrc -c IBM.TieBreaker
```

Output:

```
Resource Class Persistent Attributes for: IBM.TieBreaker
        resource 1:
                 AvailableTypes ={["SCSI",""],["EXEC",""],["Operator",""],
["Fail",""]}
```

List the tiebreaker name:

```
lsrsrc IBM.TieBreaker
```

Output:

```
Resource Persistent Attributes for: IBM.TieBreaker
        resource 1:
            Name                = "FAIL"
            Type                = "FAIL"
            DeviceInfo          = ""
            ReprobeData         = ""
            ReleaseRetryPeriod  = 0
            HeartbeatPeriod     = 0
            PreReserveWaitTime  = 0
            PostReserveWaitTime = 0
            NodeInfo            = {}

        resource 2:
            Name                = "Operator"
            Type                = "Operator"
            DeviceInfo          = ""
            ReprobeData         = ""
            ReleaseRetryPeriod  = 0
            HeartbeatPeriod     = 0
            PreReserveWaitTime  = 0
            PostReserveWaitTime = 0
            NodeInfo            = {}

resource 3:
           Name                = "myTieBreaker"
           Type                = "SCSI"
           DeviceInfo          = "ID=0 LUN=0 CHAN=0 HOST=2"
           ReprobeData         = ""
           ReleaseRetryPeriod  = 0
           HeartbeatPeriod     = 5
           PreReserveWaitTime  = 0
           PostReserveWaitTime = 0
           NodeInfo            = {}

resource 4:
           Name                = "mytb"
           Type                = "EXEC"
           DeviceInfo          = "PATHNAME=/usr/sbin/rsct/bin/samtb_net
                                          Address=192.168.177.2"
           ReprobeData         = ""
           ReleaseRetryPeriod  = 0
           HeartbeatPeriod     = 30
           PreReserveWaitTime  = 0
           PostReserveWaitTime = 30
           NodeInfo            = {}
           ActivePeerDomain    = "21"
```

Although you can define several tiebreaker resources in the resource class `IBM.TieBreaker`, only one of them can be active in the cluster at the same time. Enter the following command to list the tiebreaker that is active in the cluster:

```
lsrsrc -c IBM.PeerNode OpQuorumTieBreaker
```

Output:

```
Resource Class Persistent Attributes for: IBM.PeerNode
        resource 1:
          OpQuorumTieBreaker = "Operator"
```

Set the active tiebreaker:

```
chrsrc -c IBM.PeerNode OpQuorumTieBreaker="Operator"
```

Enter the following command to grant or deny the operational quorum when tiebreaker is Operator:

```
runact -c IBM.PeerDomain ResolveOpQuorumTie Ownership=1 (0 to deny)
```

Note: To avoid race conditions, the operator tiebreaker must be denied for the losing subcluster. Then the operator tiebreaker can be granted to the subcluster, which is supposed to continue.

Shared disk tiebreaker

Set up a disk tiebreaker in a cluster that has an even number of nodes. The tie breaker disk is shared between all cluster nodes.

A disk can be used as tiebreaker resource by using the `IBM.TieBreaker` resource class. In case only half of the number of nodes are online in a sub-domain, System Automation for Multiplatforms tries to reserve the tiebreaker disk by using the reserve or release function. If the reserve is successful, the sub-domain gets the quorum, and System Automation for Multiplatforms might continue to automate resources. The reservation of the disk is released when another node joins the domain, so that more than half of the nodes are online in that domain.

Note: When you define the tiebreaker, make sure the disk that you specify for the `IBM.TieBreaker` resource is not also used to store file systems.

The following three examples show how to use a tiebreaker with an ECKD, SCSI or DISK device. The tiebreaker does not need to be formatted or partitioned.

ECKD tiebreaker setup for a two-node cluster

Set up an ECKD tiebreaker on Linux on System z.

If the nodes are running under z/VM, see "ECKD tiebreaker in z/VM environments" on page 53 for further configuration implications regard the definition of an ECKD dasd to be used as tiebreaker.

The ECKD tiebreaker type is specific for Linux on System z. If you want to create an ECKD tiebreaker object, you need to set the `DeviceInfo` persistent resource attribute to indicate the ECKD device number. This type of tiebreaker uses a reserve or release mechanism and needs to be re-reserved periodically to hold the reservation. For this reason, you can also specify the `HeartbeatPeriod` persistent resource attribute when you create a tiebreaker of this type. The `HeartbeatPeriod` persistent resource attribute defines the interval at which the reservation request is entered again.

Collect the following system information (Linux kernel v2.4):

```
node01:~ # cat /proc/subchannels
Device sch.  Dev Type/Model CU  in use  PIM PAM POM CHPIDs
----------------------------------------------------------------------
50DE   0A6F  3390/0A   3990/E9          F0  A0  FF  7475E6E7 FFFFFFFF
```

```
node01:~ # cat /proc/dasd/devices
50dc(ECKD) at ( 94:  0) is            : active at blocksize: 4096, 601020 blocks, 2347 MB
50dd(ECKD) at ( 94:  4) is            : active at blocksize: 4096, 601020 blocks, 2347 MB
50de(ECKD) at ( 94:  8) is            : active at blocksize: 4096, 601020 blocks, 2347 MB
50df(ECKD) at ( 94: 12) is            : active at blocksize: 4096, 601020 blocks, 2347 MB
```

For Linux kernel v2.6, use the **`lscss`** command instead of the `cat /proc/subchannels` command. Perform the following steps to use the tiebreaker:

1. Create a tiebreaker resource object in `IBM.TieBreaker class`. `DeviceInfo` shows the ECKD device number. It can be obtained from `/proc/dasd/devices` file.

   ```
   node01:~ # mkrsrc IBM.TieBreaker Name=myTieBreaker \
      Type=ECKD DeviceInfo="ID=50de" HeartbeatPeriod=5
   ```

   ```
   node01:~ # lsrsrc IBM.TieBreaker
   Resource Persistent Attributes for: IBM.TieBreaker
   resource 1:
           Name                = "Operator"
           Type                = "Operator"
           DeviceInfo          = ""
           ReprobeData         = ""
           ReleaseRetryPeriod  = 0
           HeartbeatPeriod     = 0
           PreReserveWaitTime  = 0
           PostReserveWaitTime = 0
           NodeInfo            = {}
   resource 2:
           Name                = "Fail"
           Type                = "Fail"
   ```

```
        DeviceInfo          = ""
        ReprobeData         = ""
        ReleaseRetryPeriod  = 0
        HeartbeatPeriod     = 0
        PreReserveWaitTime  = 0
        PostReserveWaitTime = 0
        NodeInfo            = {}
resource 3:
        Name                = "myTieBreaker"
        Type                = "ECKD"
        DeviceInfo          = "ID=50de"
        ReprobeData         = ""
        ReleaseRetryPeriod  = 0
        HeartbeatPeriod     = 5
        PreReserveWaitTime  = 0
        PostReserveWaitTime = 0
        NodeInfo            = {}
```

2. Change `OpQuorumTieBreaker` attribute in `IBM.PeerNode` class to one of the tiebreaker resource objects.

```
node01:~ # chrsrc -c IBM.PeerNode OpQuorumTieBreaker="myTieBreaker"
```

```
node01:~ # lsrsrc -c IBM.PeerNode
Resource Class Persistent Attributes for: IBM.PeerNode
resource 1:
        CommittedRSCTVersion  = ""
        ActiveVersionChanging = 0
        OpQuorumOverride      = 0
        CritRsrcProtMethod    = 1
        OpQuorumTieBreaker    = "myTieBreaker"
```

Manually rebooting a node

If one node of a two-node cluster is rebooted, the rebooting node can come back quickly. Rebooting can disrupt the tiebreaker method and cause an undesired reboot of the remaining node. If a node that belongs to a cluster must be manually rebooted, use the command **`halt -nf`** instead of **`reboot -nf`**.

Manually breaking a disk reservation

If the node that reserves a tiebreaker is down and cannot be rebooted, manual access to the healthy node is needed to break the reservation and take it over on that node.

- The tiebreaker disk can either be still attached to the healthy node, provided this node has not been rebooted in the mean time:

```
node01:~ # cat /proc/subchannels
Device sch. Dev Type/Model CU in use PIM PAM POM CHPIDs
------------------------------------------------------------
50DE 0A6F 3390/0A 3990/E9 F0 A0 FF 7475E6E7 FFFFFFFF

node01:~ # cat /proc/dasd/devices
50de(ECKD) at ( 94: 8) is dasdc: active at blocksize: 4096,601020 blocks, 2347 MB
```

- The tiebreaker disk can be boxed if this node is rebooted and cannot recognize the tiebreaker disk anymore:

```
node01:~ # cat /proc/subchannels
Device sch. Dev Type/Model CU in use PIM PAM POM CHPIDs
------------------------------------------------------------
50DE   0A6F            FFFF/00          F0  A0  FF  7475E6E7 FFFFFFFF

node01:~ # cat /proc/dasd/devices
50de(ECKD) at ( 94: 8) is dasdc : boxed
```

  To break the tiebreaker disk reservation enter the command `/usr/sbin/rsct/bin/tb_break`:

```
tb_break -t ECKD /dev/dasdc
```

  The tiebreaker disk is now reserved by the healthy node.

Note: If the **tb_brk** command does not work the first time, enter it again.

SCSI tiebreaker setup for a two-node cluster

Set up an SCSI tiebreaker on Linux on System x or Linux on POWER.

This SCSI tiebreaker type is specific for Linux on System x, and Linux on POWER. If you want to create a SCSI tiebreaker object, you must specify the SCSI device with the `DeviceInfo` persistent resource attribute. If the SCSI configuration is different on different nodes in the cluster, you can also use the `NodeInfo` persistent resource attribute to reflect those differences. This type of tiebreaker uses a reserve/release mechanism and must be re-reserved periodically to hold the reservation. When you create a tiebreaker of this type, you can also specify the `HeartbeatPeriod` persistent resource attribute. The `HeartbeatPeriod` persistent resource attribute defines the interval at which the reservation request is reissued.

SCSI devices on Linux can be identified by four integer values for the attributes HOST, CHAN, ID, and LUN:

```
node1:~# dmesg | grep "Attached scsi disk"
```

Normally these parameters are identical on each cluster node. For example, for node1 and node2 the parameters are HOST=0 CHAN=0 ID=4 LUN=0.

In this case use the following command to create the tiebreaker object:

```
mkrsrc IBM.TieBreaker Name=myTieBreaker Type=SCSI DeviceInfo=" HOST=0 CHAN=0
 ID=4 LUN=0"
```

The four values can also be different for different nodes (even if the target device is same). In that case, use the `NodeInfo` field in addition to the `DeviceInfo` field.

Use the four integer values from the command output:

```
# dmesg | grep "Attached scsi disk"
Attached scsi disk sdf at scsi2, channel 2, id 4, lun 0
```

For disk `sdf` the values of the SCSI identifier attributes are HOST=2, CHAN=2, ID=4, LUN=0. For example, a SCSI device is connected to two nodes that are named node1 and node2 and has the following SCSI identifiers:

```
node1:  HOST=0 CHAN=0 ID=4 LUN=0
node2:  HOST=2 CHAN=2 ID=4 LUN=0
```

Create the tiebreaker object by using `DeviceInfo` to specify common attribute values and `NodeInfo` to specify node-specific attribute values:

```
# mkrsrc IBM.TieBreaker Name=scsi Type=SCSI DeviceInfo="ID=4 LUN=0"
NodeInfo='{["node1", "HOST=0 CHAN=0"], ["node2", "HOST=2 CHAN=2"]}'
```

System Automation for Multiplatforms handles `DeviceInfo` and `NodeInfo` in such a way that it merges the two strings, `DeviceInfo` first and then `NodeInfo`. For example, for node1 the merged string is:

```
"ID=4 LUN=0 HOST=0 CHAN=0"
```

This string is parsed.

Also, any duplicated keywords are allowed and the last one is used. Therefore, the same command can be specified as

```
# mkrsrc IBM.TieBreaker Name=myTieBreaker Type=SCSI DeviceInfo="ID=4 LUN=0
HOST=0,CHAN=0"  NodeInfo='{["node2", "HOST=2 CHAN=2"]}'
```

This simplification can be useful as in most cases the SCSI id is the same for many nodes.

Manually breaking a disk reservation

If the node that reserves a tie breaker is down and cannot be rebooted, manual access to the healthy node is needed to release the SCSI tiebreaker disk. To release a disk, run the **tb_break [-f] HOST CHAN ID LUN** command, for example

```
/usr/sbin/rsct/bin/tb_break -f HOST=0 CHAN=0 ID=4 LUN=0
```

AIX DISK tiebreaker setup for a two-node cluster

Set up an AIX DISK tiebreaker on AIX systems.

The DISK tiebreaker type is specific to AIX. If you want to create a DISK tiebreaker object, you need to set the `DeviceInfo` persistent resource attribute to indicate the AIX device name. The AIX device name must specify a SCSI or SCSI-like physical disk that is shared by all nodes of the peer domain.

Physical disks that are attached via Fibre Channel, iSCSI, and Serial Storage Architecture might serve as a DISK tiebreaker. IDE hard disks do not support the SCSI protocol and cannot serve as a DISK tiebreaker. Logical volumes also cannot serve as a DISK tiebreaker. This type of tiebreaker uses a reserve or release mechanism and needs to be re-reserved periodically to hold the reservation. For this reason, you can also specify the `HeartbeatPeriod` persistent resource attribute when you create a tiebreaker of this type. The `HeartbeatPeriod` persistent resource attribute defines the interval at which the reservation request is entered again.

Use the following command to list every known physical volume in the system along with its physical disk name:

```
lspv
```

An output similar to the following one is displayed:

```
hdisk0 000000371e5766b8 rootvg active
hdisk1 000069683404ed54 None
```

Use the **lsdev** command to verify that a disk is a SCSI or SCSI-like disk. This disk is a suitable candidate for a DISK tiebreaker. For example:

```
lsdev -C -l hdisk1
```

An output similar to the following one is displayed:

```
hdisk1 Available 10-60-00-0,0 16 Bit SCSI Disk Drive
```

To serve as a tiebreaker disk, the disk must be shared by all nodes of the peer domain. Check the physical volume ID returned by the **lspv** command to determine, if the disk is shared between nodes. In the preceding output for the **lspv** command, the physical volume ID is listed in the second column; the volume ID for hdisk1 is 000069683404ed54. AIX remembers all disks that are attached to the system, and the disks that are listed by the **lspv** command can no longer be attached. If such a disk was moved to another system, it might appear as if the disk is shared, but it is no longer attached to the original system.

Make sure that the disk on which `IBM.TieBreaker` resources are stored do not also store file systems. If the nodes of the cluster share more than one disk, it can be difficult to determine which disk is the tiebreaker disk and, which one is used for application data. The output from the **lsdev** command shows the SCSI address that is associated with the disk. (In the preceding output for the **lsdev** command, the SCSI address is listed in the third column; the SCSI address for hdisk0 is 10-60-00-0,0). This information helps you to identify the correct disk if you know the address of the disk before its installation.

After you determined the device name, use the **mkrsrc** command to define the tiebreaker object:

```
mkrsrc IBM.TieBreaker Name=myTieBreaker \
Type=DISK DeviceInfo="DEVICE=/dev/hdisk1" HeartbeatPeriod=5
```

Checking the SCSI reservation capability

The tiebreaker relies on SCSI-2 reservation, which is not necessarily supported by every combination of storage and driver setup. To check that the setup supports SCSI-2 reservation, RSCT provides the **disk_reserve** utility that must be started with its full path `/usr/sbin/rsct/bin/disk_reserve`.

The tiebreaker works correctly if the tiebreaker disk can be reserved and unlocked from either node and if the disk cannot be reserved from a node while it is locked by the other node.

Usage:

```
/usr/sbin/rsct/bin/disk_reserve [-l | -u | -b] [-h] [-v] [-f] [-d sdisk_name]
/usr/sbin/rsct/bin/disk_reserve [-l | -u | -b] [-h] [-v] [-f] [-g sg_device_name]
```

`-h` - display this help text

`-v` - verbose

`-f` - reserve after break (for the `-l` or `-b` option)

`-d sdisk_name` - disk to operate, for example `/dev/sdb`

`-l` - lock (reserve)

`-u` - unlock (release)

`-b` - break

`-g sg_device_name` , for example `/dev/sg1`

Examples:

```
/usr/sbin/rsct/bin/disk_reserve -l -f -d /dev/sde
/usr/sbin/rsct/bin/disk_reserve -l -g /dev/sg3
```

Manually breaking a disk reservation

If the node that reserves a tiebreaker is down and cannot be rebooted, manual access to the healthy node is needed to release the SCSI tiebreaker disk. To release the disk, use the **tb_break** command, for example:

```
/usr/sbin/rsct/bin/tb_break –f –t DISK "DEVICE=/dev/hdisk1"
```

The following is an example for a disk that does not satisfy the criteria to serve as a tie breaker disk. Enter the **lspath** command, for example:

```
lspath -l hdisk2
lspath: 0514-538 Cannot perform the requested function because the
                 specified device does not support multiple paths.
```

Sample output:

```
#lspath -l hdisk2
Enabled hdisk2 fscsi0
Failed  hdisk2 fscsi0
Failed  hdisk2 fscsi0
Failed  hdisk2 fscsi0
Failed  hdisk2 fscsi0
Enabled hdisk2 fscsi0
Enabled hdisk2 fscsi0
Enabled hdisk2 fscsi1
Failed  hdisk2 fscsi1
Failed  hdisk2 fscsi1
Failed  hdisk2 fscsi1
Failed  hdisk2 fscsi1
```

```
Enabled hdisk2 fscsi1
Enabled hdisk2 fscsi1
```

This sample output shows that the disk does not support SCSI-2 reservation and cannot be used as a tiebreaker.

SCSI persistent reserve for the disk tiebreaker

You can configure a disk tiebreaker to use SCSI persistent reserve on AIX and Linux for System x. Starting with System Automation for Multiplatforms version 3.2.1.3, this functionality is extended to include Linux for System z.

SCSI-3 tiebreaker on AIX

By default, the tiebreaker of type DISK on AIX relies on SCSI-2 reserve or release, which is not necessarily supported by every combination of SCSI disk storage and driver setup. Typically, storage virtualization solutions like SAN Volume Controller do not support SCSI-2 reservation. In these environments, the AIX operating system can be configured to transform SCSI-2 reserve or release commands to SCSI-3 persistent reserve commands.

Use the following command to configure SCSI-2 reserve or release to SCSI-3 persistent reserve transformation on AIX:

```
chdev -l <pv_name> -a PR_key_value=0x<unique_key> -a reserve_policy=PR_exclusive
```

<pv_name>
The name of the physical volume on the AIX system to be used for tie breaking.

<unique_key>
Is an arbitrary numeric key that is unique to each node in the cluster.

Run this command on each remote peer system of the domain and specify a different unique key on each system. To find out whether a SCSI disk to be used for DISK tiebreaker supports this approach, run

```
lsattr -El <pv_name>
```

Look for the attributes `PR_key_value` and `reserve_policy`. If the attributes cannot be adjusted as described in the preceding paragraphs, check for missing device drivers at Host Attachment for SDDPCM on AIX.

Disks on POWER blades within a zBX environment can be only defined as `Virtual SCSI Disk drive`. They cannot be configured to support SCSI-2 reserve or release, or SCSI-3 persistent reserve. Therefore, these disks cannot be used as disk tiebreaker.

SCSIPR tiebreaker on Linux for System x

System Automation for Multiplatforms version 3.2.1.2 introduced the tiebreaker type SCSIPR, which is specific to Linux on System x. It is supported on RHEL 7, RHEL 8, SLES 12 and SLES 15.

The SCSIPR tiebreaker uses SCSI-3 persistent reservations on a SCSI disk storage device as tie breaking mechanism. If a tie situation in which the peer domain is partitioned into two subdomains and each subdomain contains exactly half of the defined node, then the subdomain, which is able to get an exclusive persistent reservation of the shared SCSI disk storage device obtains the operational quorum.

Prerequisites

The SCSI disk storage device to be used by SCSIPR tiebreaker must support the SCSI-3 persistent reserve protocol with reservation type `Write Exclusive Registrants Only`. This device must be shared by all systems in the peer domain and all systems must be able to reserve the device by using the SCSI-3 persistent reserve protocol.

The SCISPR tie breaker uses the `sg_persist` utility. Use the following commands to check whether it is already installed on all systems of the peer domain:

```
which sg_persist
rpm -qf /usr/bin/sg_persist
```

If the `sg_persist` utility is not installed yet, you need to install the appropriate Linux package:

- RHEL 7, RHEL 8, SLES 12 & SLES 15: `sg3_utils*.rpm`

Definition

When you create a tiebreaker of type SCSIPR, use the `DeviceInfo` persistent resource attribute to specify the SCSI disk storage device to be used by the tiebreaker. If the SCSI configuration is different between peer domain systems, use the `NodeInfo` persistent resource attribute to reflect those differences.

The SCSIPR tiebreaker uses a reserve or release mechanism and needs to be reserved again periodically to hold the reservation. For this reason, specify the `HeartbeatPeriod` persistent resource attribute when you create a tiebreaker of this type. The `HeartbeatPeriod` persistent resource attribute defines the interval at which the reservation is retried.

Note: When you define tiebreaker resources, be aware that the disk on which `IBM.Tiebreaker` resources are stored is not also used to store file systems.

Use one of the following options to identify the SCSI disk storage device to be used by the tiebreaker in the `DeviceInfo` persistent resource attribute:

- `DEVICE=<generic or disk device name>`
- `HOST=<h> CHAN=<c> ID=<i> LUN=<I>`
- `WWID=<wwid as displayed by the system>`
- `RDAC.WWN=<wwn as displayed by the system when using LSI RDAC multi-path driver>`

Example:

```
mkrsrc IBM.TieBreaker Name="mySCSIPRTieBreaker" Type=SCSIPR DeviceInfo="DEVICE=/dev/sdx"
HeartbeatPeriod=5
```

Verification

Perform the following steps on all remote peer systems to verify whether all systems support the SCSIPR tiebreaker correctly with the chosen SCSI disk storage device:

- Reserve the disk device by using the **tb_break** command:

```
/usr/sbin/rsct/bin/tb_break -l -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must be able to successfully reserve the disk device.

- Try to reserve the same disk device by using the **tb_break** command on all other peer domain systems:

```
/usr/sbin/rsct/bin/tb_break -l -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must fail to reserve the disk device because it is already exclusively reserved by the first system.

- Release the disk device by using the **tb_break** command:

```
/usr/sbin/rsct/bin/tb_break -u -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must be able to successfully release the disk device.

Checking whether a reservation is held:

Use the following command to check whether a reservation is held on the SCSI disk storage device:

```
sg_persist --read-reservation <generic or disk device name>
```

Example: no reservation is held:

```
sg_persist --read-reservation /dev/sdx
IBM 2145 0000
Peripheral device type: disk
PR generation=0x5, there is NO reservation held
```

Example: reservation is held:

```
sg_persist --read-reservation /dev/sdx
IBM 2145 0000
Peripheral device type: disk
PR generation=0x5, Reservation follows:
Key=0x11293693fa4d5807
scope: LU_SCOPE, type: Write Exclusive, registrants only
```

When you reserve a disk device, each remote peer system uses its RSCT node identifier as reservation key. A remote peer system's RSCT node identifier can be displayed by using the **/usr/sbin/rsct/bin/lsnodeid** command. If a SCSI disk storage device is reserved by the SCSIPR tiebreaker, you can determine the system that is holding the reservation. Determine the current reservation key and compare it to all remote peer systems RSCT node identifier.

Breaking a reservation:

If a remote peer system currently holds a reservation on the SCSI disk storage device, it is possible to break this reservation from another remote peer system. Use the following command to forcefully break an existing reservation and obtain a new reservation:

```
/usr/sbin/rsct/bin/tb_break -f -t SCSIPR <DeviceInfo device specification for this system>
```

SCSIPR tiebreaker on Linux for System z

System Automation for Multiplatforms version 3.2.1.3 introduced the tiebreaker type SCSIPR for use with Linux on System z. It is supported on SLES 12 and SLES 15.

The SCSIPR tiebreaker uses SCSI-3 persistent reservations on a SCSI disk storage device as tie breaking mechanism. If a tie situation in which the peer domain is partitioned into two subdomains and each subdomain contains exactly half of the defined node, then the subdomain, which is able to get an exclusive persistent reservation of the shared SCSI disk storage device obtains the operational quorum.

Prerequisites

The SCSI disk storage device to be used by SCSIPR tiebreaker must support the SCSI-3 persistent reserve protocol with reservation type `Write Exclusive Registrants Only`. This device must be shared by all systems in the peer domain and all systems must be able to reserve the device by using the SCSI-3 persistent reserve protocol. The SCISPR tiebreaker uses the `sg_persist` utility. Use the following commands to check whether it is already installed on all systems of the peer domain:

```
which sg_persist
rpm -qf /usr/bin/sg_persist
```

If the `sg_persist` utility is not installed yet, you need to install the appropriate Linux package:

- RHEL 7, RHEL 8, SLES 12, and SLES 15: `sg3_utils*.rpm`

The disk that serves as tiebreaker must have N port identifier virtualization enabled. Otherwise, each reservation is run on behalf of the entire CEC, the zSeries physical box, instead of a single logical partition on that CEC. For more information about N port identifier virtualization on zSeries, refer to:

- Redpaper: Introducing N_Port Identifier Virtualization for IBM System z9®

- Redbooks: Fibre Channel Protocol for Linux and z/VM on IBM System z

Definition

When you create a tiebreaker of type SCSIPR, use the `DeviceInfo` persistent resource attribute to specify the SCSI disk storage device to be used by the tiebreaker. If the SCSI configuration is different between peer domain systems, use the `NodeInfo` persistent resource attribute to reflect those differences.

The SCSIPR tiebreaker uses a reserve or release mechanism and needs to be reserved again periodically to hold the reservation. For this reason, specify the `HeartbeatPeriod` persistent resource attribute when you create a tiebreaker of this type. The `HeartbeatPeriod` persistent resource attribute defines the interval at which the reservation is tried again.

Note: When you define tiebreaker resources, be aware that the disk on which `IBM.Tiebreaker` resources are stored are not used to store file systems.

Use one of the following options to identify the SCSI disk storage device to be used by the tiebreaker in the `DeviceInfo` persistent resource attribute:

- `DEVICE=<generic or disk device name>`
- `HOST=<h> CHAN=<c> ID=<i> LUN=<l>`
- `WWID=<wwid as displayed by the system>`
- `RDAC.WWN=<wwn as displayed by the system when using LSI RDAC multi-path driver>`

Example:

```
mkrsrc IBM.TieBreaker Name="mySCSIPRTieBreaker" Type=SCSIPR DeviceInfo="DEVICE=/dev/sdx"
HeartbeatPeriod=5
```

Verification

Perform the following steps on all remote peer systems to verify whether all systems support the SCSIPR tie breaker correctly with the chosen SCSI disk storage device:

- Reserve the disk device by using the `tb_break` command:

```
/usr/sbin/rsct/bin/tb_break -l -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must be able to successfully reserve the disk device.
- Try to reserve the same disk device by using the `tb_break` command on all other peer domain systems:

```
/usr/sbin/rsct/bin/tb_break -l -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must fail to reserve the disk device because it is already exclusively reserved by the first system.
- Release the disk device by using the `tb_break` command:

```
/usr/sbin/rsct/bin/tb_break -u -t SCSIPR <DeviceInfo device specification for this system>
```

  This command must be able to successfully release the disk device.

Checking whether a reservation is held:

Use the following command to check whether a reservation is held on the SCSI disk storage device:

```
sg_persist --read-reservation <generic or disk device name>
```

Example: no reservation is held:

```
sg_persist --read-reservation /dev/sdx
IBM 2145 0000
Peripheral device type: disk
PR generation=0x5, there is NO reservation held
```

Example: reservation is held:

```
sg_persist --read-reservation /dev/sdx
IBM 2145 0000
Peripheral device type: disk
PR generation=0x5, Reservation follows:
Key=0x11293693fa4d5807
scope: LU_SCOPE, type: Write Exclusive, registrants only
```

When you reserve a disk device, each remote peer system uses its RSCT node identifier as reservation key. A remote peer system's RSCT node identifier can be displayed by using the **/usr/sbin/rsct/bin/lsnodeid** command. If a SCSI disk storage device is reserved by the SCSIPR tiebreaker, you can determine the system that is holding the reservation by determining the reservation key. Compare the reservation key to all RSCT node identifier of the remote peer systems.

Breaking a reservation:

If a remote peer system holds a reservation on the SCSI disk storage device, it is possible to break this reservation from another remote peer system. Use the following command to forcefully break an existing reservation and obtain a new reservation:

```
/usr/sbin/rsct/bin/tb_break -f -t SCSIPR <DeviceInfo device specification for this system>
```

ECKD tiebreaker in z/VM environments

On Linux on System z®, an ECKD™ DASD can be used as tiebreaker resource.

The ECKD tiebreaker uses the reserve and release function, which can lead to additional configuration steps. A reserved ECKD DASD cannot be accessed by the z/VM®. Therefore, z/VM cannot attach or vary online such a device that is reserved by another system. To work around this situation, a set of configuration actions is required. The corresponding requirements are explained in the following sections.

ECKD DASD requirements for domains that run in a single z/VM system

If all nodes of the System Automation domain are guests of the same z/VM system, the following definitions are required for the ECKD DASD:

- A full pack MiniDisk is defined.
- If MiniDisk cache is used, its value is set to `off`.
- The ECKD DASD is shared between both guests in the z/VM system.

ECKD DASD requirements for domains that span two z/VM systems

If the nodes of the System Automation domain are guests of two different z/VM systems, the following definitions are required for the ECKD DASD:

- The tiebreaker disk needs to be defined as a DEVNO disk in a MiniDisk statement in the user profile (no MiniDisk, no fullpack MiniDisk, no dedicated or attached DASD)
- The ECKD disk (DEVNO) is shared between both nodes.
- The ECKD DASD must not be system that is attached when the z/VM is IPL'ed

Logging on to the Linux guests shows the following device attachment, a virtual device (291 in the example) with the real address (4a82 in the example). The device becomes shared in the example with the command `cp set shared on 4a82`. The device needs to be shared on both sides.

```
00: CP Q 4A82
00: DASD 4A82 CP SYSTEM DEVNO   1     SHARED
00:
00: CP Q V 291
00: DASD 0291 3390 VM4A82 R/W      3339 CYL ON DASD 4A82 SUBCHANNEL = 000F
```

In case one of the z/VM systems is shut down, the ECKD DASD is reserved by the surviving Linux guest on the other z/VM system. On the surviving side, you can see the following output:

```
00: CP Q DA RESERVE
00: DASD 4A82 CP SYSTEM DEVNO    1     RESERVED BY USER test1
```

After you start the z/VM system again, the DASD 4A82 is still offline and cannot be set online, because it is still reserved by the other system. A timeout of 20 – 30 minutes occurs instead.

The recommendation is to start Linux on the restarted z/VM without the tiebreaker DASD. This succeeds, since the DASD is not needed for starting Linux. After Linux is started, System Automation will start automatically on the Linux guest, and then Linux will automatically join the System Automaton domain again. The reservation of the ECKD DASD is released. It is possible to vary on the device of the tiebreaker disk (4a82 in the example). Commit the **share** command and link the virtual address of the tiebreaker disk (291 in this example) on the newly IPL'ed system. Enter the command **chccwdev -e 291** on the restarted Linux. After this command completed, everything is up and running. No further interaction on the surviving Linux is necessary.

All required commands are CP commands. Therefore, a script that processes those commands by using VMCP can be written to automate the restoration of the failed Linux.

For the example, the script might contain the following commands:

```
vmcp vary on 4a82
vmcp set shared on 4a82
vmcp link * 291 291 mr
chccwdev -e 291
```

System Automation recognizes the newly defined DASD automatically.

Network tiebreaker

The network tiebreaker provides an alternative to the disk and operator-based tie breakers. It uses an external IP (network instance) to resolve a tie situation.

There are several situations in which the use of a network tiebreaker is most appropriate, for example:

- A shared disk to be used as a disk tiebreaker is not available.
- The ability to communicate with instances outside the cluster has the highest priority.

Example: The primary function of a web server is to deliver web pages to clients outside of the cluster. To make this service highly available, the tiebreaker must not grant access to a node, which is not able to communicate to instances outside of the cluster.

Use the network tiebreaker only for domains where all nodes are in the same IP sub net. Having the nodes in different IP sub nets makes it more likely that both nodes can ping the network tiebreaker, while they cannot communicate with each other. Additionally, the default gateway IP address must not be used if it is virtualized by the network infrastructure. Choose an IP address, which can be reached only through a single path from each node in the domain.

In the default setting, the network tiebreaker makes two attempts to ping the network tie breaker IP address. This default number of pings can be too low in virtualized environments or environments with a slow or unreliable network connection. For those environments, you can increase the number of pings that are executed by the network tie breaker up to a maximum of nine. Then, you can ensure a correct result of the tie breaker reserve operation.

Requirements for the network tiebreaker

To ensure the network tiebreaker function, the external IP instance must be reachable from all nodes within the highly available cluster. Also, the external IP instance must be able to reply to ICMP echo requests (ping). If you install a firewall rule, which blocks ICMP traffic between the cluster nodes and the external IP instance, the network tiebreaker does not work. In this situation, the cluster nodes might not communicate to their peers (cluster split), but both sub clusters are able to reach the external IP

instance. Usually, IP ensures that if both sub clusters can reach the external gateway, they are also able to communicate with their peers. If this rule cannot be ensured, for example due to firewall settings, you cannot use the network tiebreaker.

The following table shows the advantages and disadvantages of network and disk tie breakers:

Table 20. Comparison of network-based and disk-based tie breakers

Network-based tiebreaker	**Disk-based tiebreaker**
• +: No hardware dependency. • +: Evaluates the availability of communication.	• +: Most secure tiebreaker. Hardware ensures that only one instance (node) is able to get the tiebreaker.
• -: If the external IP instance is not available in case of a cluster split, no sub-cluster gets quorum. • -: There can be error conditions in which a tie situation occurs, but more than one node is able to communicate. In this case, both sub-clusters are able to get the tiebreaker.	• -: If there is a loss in communication, this tiebreaker can grant access to a node, which is not able to communicate to instances outside the cluster.

Setting up a network tiebreaker

Define a network tiebreaker as an `IBM.TieBreaker` resource of type EXEC. For more information about an EXEC tie breaker, see the RSCT documentation. The network tiebreaker executable files `samtb_net` and `samtb_net6` are in the `/usr/sbin/rsct/bin` directory. In the current implementation, the following options must be specified as `key=value` pairs during the creation of the RSCT EXEC tiebreaker:

`Address=<IP address>`
Address of the external IP instance, which is used to resolve the tie situation. Within an IPv6 network, specify an address in IPv6 format. Do not use a DNS name. DNS can not work properly if communication problems occur, which is typically the case during cluster splits. Address is a mandatory option, there is no default value.

`Log=<1/0>`
Specify 1 if you want the network tiebreaker to write logs to the system log facility (syslog). Otherwise, specify 0.

`Count=<number>`
Number of ICMP echo requests, which are sent to request quorum. If the first request gets a response, no further requests are sent. The default value is 2. The allowed value range is 1 - 9. Increase the value for Count for virtual environments or environments with a slow or unreliable network connection.

Depending on the IP version, there are different network tiebreaker executable files, that you must use when you define the tiebreaker.

The following command creates a new network tiebreaker for an IPv4 address:

```
# mkrsrc IBM.TieBreaker Type="EXEC" Name="mynetworktb" \
DeviceInfo='PATHNAME=/usr/sbin/rsct/bin/samtb_net Address=<IPv4 address> \
Log=1' PostReserveWaitTime=30;
```

The following command creates a new network tiebreaker for an IPv6 address:

```
# mkrsrc IBM.TieBreaker Type=EXEC Name="mynetworktb" \
DeviceInfo='PATHNAME=/usr/sbin/rsct/bin/samtb_net6 Address=<IPv6 address> \
Log=1' PostReserveWaitTime=30;
```

Activate your network tiebreaker as follows:

```
# chrsrc -c IBM.PeerNode OpQuorumTieBreaker="mynetworktb"
```

Use the **chrsrc** command to manipulate the network tiebreaker definition. For example, if you want to increase the value for the number of pings, enter the following commands:

```
chrsrc -c IBM.PeerNode OpQuorumTieBreaker="Operator"
chrsrc -s "Name = 'your_tiebreaker_name'" IBM.TieBreaker \
DeviceInfo="PATHNAME=/usr/sbin/rsct/bin/samtb_net Address=<network-tb-ip> \
Count=<new-value-for-Count> Log=1"
chrsrc -c IBM.PeerNode OpQuorumTieBreaker="your_tiebreaker_name"
```

To delete the tiebreaker definition, use the **rmrsrc** command.

Reserve behavior of a network tiebreaker

When a node reserves a tiebreaker, the tiebreaker is no longer available and cannot be reserved by any other node. This approach is not feasible for a network tiebreaker. Therefore, the reserve behavior of a network tiebreaker is different in the following way.

After an unsuccessful reservation attempt, no other reservation is possible until the node joined the cluster again. A file is written to `/var/ct/`, which indicates that a reservation failed. If this file is present, a reserve command for a tiebreaker always fails. An extra process is forked, which watches quorum and removes the block file if the node joined the domain again.

The following sample file was created by the network tiebreaker as the result of a failing tie breaker reserve operation to the external IP instance 123.456.789.1. It contains the time stamp of the failed reserve operation.

```
# cat /var/ct/samtb_net_blockreserve_123.456.789.1
Mo Jul 4 08:38:40 CEST 2005
```

Configuring a tiebreaker resource for the network tie breaker

This topic describes the tiebreaker configuration options that need to be considered, when you define a network tiebreaker.

PostReserveWaitTime=30
: The `PostReserveWaitTime` defines the delay between the point in time where the tiebreaker is successfully reserved and the point in time where quorum is granted. A node that reserves the network tiebreaker does not get operational quorum until the `PostReserveWaitTime` is passed. Specify a value of 30 seconds to provide enough outage time. This time is needed by the nodes to detect that the other node is offline and to immediately restore the communication. In this case, both nodes are able to reserve the network tiebreaker. Due to the longer wait time, the communication between the nodes is established again and the chance that both nodes get quorum and start resources in parallel is minimized.

HeartbeatPeriod=30
: After there was a successful reserve, `ConfigRM` starts calling periodically the tiebreaker heartbeat operation. To keep the system load low during a cluster split, increase the time between the tiebreaker heartbeats or even turn off heart beating by setting `HeartbeatPeriod` to 0.

Reviewing the system logs of a network tiebreaker

The following shows a sample system log content for a network tiebreaker error scenario in a two-node cluster (node n1 and node n2).

In Figure 12 on page 57, you can see the system logs of a two-node cluster (node n1 and node n2). For the error scenario, it is assumed that there are no critical resources that are running on both nodes. A network problem breaks all available communication paths between the peers, but one peer (n2) is still able to communicate to its gateway (123.456.789.1). After some time, communication is established again and both nodes can join the cluster.

Figure 12. Two-node cluster system logs

NFS tiebreaker

The NFS tiebreaker resolves tie situations that are based on reserve files that are stored on an NFS v4 server. The NFS server can be used for multiple System Automation for Multiplatforms clusters. If the same server is used for multiple tie NFS breakers, each tiebreaker needs a reserve file with a unique name.

It is not possible that in a cluster split situation more than one node has quorum or pending quorum at any time. If the node, which obtained the quorum fails afterward, other nodes automatically try to obtain quorum that is based on the challenger-defender protocol.

The NFS server can be on any system that supports to run NFS v4. If you use an NFS server, which is compliant with the newer v4.1 or pNFS standard for System Automation for Multiplatforms tie breakers, make sure that the replication and failover capabilities of the NFS server are disabled. Use the NFS server for System Automation for Multiplatforms tiebreaker purposes only.

NFS v4 client libraries must be installed on all System Automation for Multiplatforms cluster nodes.

An example scenario for using an NFS tiebreaker is a three site setup. Two sites host a set of two-node clusters and the tiebreaker is supposed to be on the third site. A disk tiebreaker cannot be used, because it requires a SAN setup that is not necessarily crossing all three sites. It is also not possible to make any assumptions about the network topology. No network device on the third site can be chosen as destination address for the network tiebreaker. In this case, the third site can be used to host the NFS v4 server that is used as tiebreaker.

If the NFS quorum server is down or not accessible in a cluster split situation, cluster nodes do not get quorum. This situation is similar to a disk tiebreaker, where no node gets quorum if the disk device failed or is unreachable. Make sure that the NFS quorum server is permanently running and works reliably.

System Automation mounts the NFS file system at various stages to the cluster nodes, but not periodically.

`Initialize`
: The mount is established when the NFS tie breaker is set as the active tie breaker, during the `Initialize` operation. The same happens during domain or node start-up. If this fails, the node might be unable to join the domain.

`Reserve`
: During the `Reserve` operation, before the reserve file is accessed, the NFS mount is checked, and (re-)established if needed.

`Terminate`
: The NFS file system is unmounted during the `Terminate` operation, which runs when the NFS tie breaker is no longer the active tie breaker, or when the domain/node is stopped.

System Automation for Multiplatforms mounts the NFS file system at various stages on the cluster nodes, but not periodically:

- Initially the mount is established when the NFS tie breaker is set as the `active tie breaker` during the `Initialize` operation or the domain or node startup. If the mount fails, the node might be unable to join the domain.
- During the `Reserve` operation before the `reserve file` is accessed, the NFS mount is checked, and (re-) established if needed.

The NFS file system is unmounted during the `Terminate` operation, which is performed when the NFS tie breaker is no longer the active tie breaker, or when the domain or node is stopped.

Note: Existence of the reserve file is crucial in case of a cluster split and deleting the reserve file can cause both nodes in a cluster to be granted quorum. Use a naming schema for these files that allows for a direct association between the reserve file and the cluster by using the reserve file. For example, `NFS_reserve_file_SAP_HA_sapnode1_sapnode2_DO_NOT_REMOVE` clearly states the purpose of the file, the name of the cluster, and the names of the nodes that use the reserve file. If the file was deleted, activate the default operator tiebreaker, create the file again, and then activate the NFS tiebreaker again. For more information about the operator tie breaker, see "Configuring the tiebreaker" on page 42.

Enabling the NFS server on Linux

Find out how to enable NFS v4 support if you are running System Automation for Multiplatforms on Linux.

Enable NFS v4 support:

1. Add the following line to `/etc/exports` for the quorum server file system:

   ```
   </your/quorumServerDir> *(fsid=0,rw,sync,no_root_squash)
   ```

   The directory name `</your/quorumserverDir>` is an example. You can use any directory name. Make sure that only one path is exported with `fsid=0`.
2. Create a directory `<quorum_server_directory>` and set its permission bit mask to `a+rwxt`.
3. You might need to add the following lines to `/etc/fstab` to mount the `rpc_pipefs` and `nfsd` file systems automatically:
   a. `rpc_pipefs /var/lib/nfs/rpc_pipefs rpc_pipefs defaults 0 0`
   b. `nfsd /proc/fs/nfsd nfsd defaults 0 0`
4. You might need to restart your server so the changes in the `config` files in the `/etc` directory are applied. Refer to the documentation of your Linux distribution for further details.
5. Verify that the directories `/var/lib/nfs/v4recovery/` and `/var/lib/nfs/ rpc_pipefs/` were created. Depending on what distribution you use, you might need to load the NFS kernel module by running the `modprobe nfs` command.
6. Depending on the distribution that you use, daemons are started in a different way. For example, you might need to type `/etc/init.d/idmapd start` or `service idmapd start` to start the `rpc.idmapd` daemon. The following daemons must be started:
   a. `rpc.idmapd`
   b. `rpc.gssd`
   c. `rpc.nfsd`
7. Refresh the export list by running the `exportfs -r` command.
8. Verify that the `rpc.nfsd` and `rpc.idmapd` daemons are up and running.
   a. `rpc.nfsd`: Use the command `ps -ef | grep nfsd` to verify that a process with the name `nfsd` is up and running.
   b. `rpc.idmapd`: Use the command `ps -ef | grep rpc.idmapd`
   c. Use the command `rpcinfo -p` to verify the versions of all registered RPC programs.

The NFS v4 ID for the mapping daemon `rpc.idmapd` is required to run on the System Automation for Multiplatforms node, which is running the NFS tiebreaker. Refer to your distributions documentation on how to start the `idmapd` daemon.

To verify that a System Automation for Multiplatforms node can correctly access the NFS server, enter the following command:

```
mount –t nfs4 <nfs_server_name>:/<quorum_directory_name>/<local_directory>
```

The installation is successfully verified, if the mount command succeeds and if it is possible to create files on the mounted NFS v4 directory.

If the mount operation does not succeed, fix your installation with the help of your operating system documentation.

Refer to the documentation of your Linux distribution for further details.

Enabling the NFS server on AIX

Find out how to enable NFS v4 support if you are running System Automation for Multiplatforms on AIX.

Make sure that the NFS v4 related daemons are started on the NFS server:

1. Verify that the NFS v4 related daemons are started on your server with the `lssrc –g nfs` command.
2. Start the NFS server by running the following commands, if the NFS server is not yet started:
   a. `mknfs`
   b. `chnfsdom <your_nfs_domain_name>`
   c. `startsrc –s nfsrgyd`
3. Create a `<quorum_server>` directory and set its permission bit mask to `a+rwxt`.
4. Export that directory to NFS v4 clients with the `mknfsexp –v 4 –d <quorum_server> [ -h <host>]` command.
5. You can restrict the list of hosts, which are allowed to mount the directory for security reasons. Restrict the list of hosts to all System Automation for Multiplatforms nodes, which use the NFS server by specifying the –h option.

Start and configure the NFS-related daemons, which are required on the NFS client by running the `mknfs` command. In case the used NFS server is running on Linux, you might see the following error message in the system log after an NFS tiebreaker is initialized:

```
vmount: operation not permited
```

The Linux NFS server checks if the port for the NFS client is a reserved port. In case you receive an error message, run the following command on every AIX system where the NFS tiebreaker runs.

```
nfso -p -o nfs_use_reserved_ports=1
```

To verify that a System Automation for Multiplatforms node can correctly access the NFS server, enter:

```
mount –o vers=4 <nfs_server_name>:/<quorum_directory_name>/<local_directory>
```

The installation is successfully verified, if the mount command succeeds and if it is possible to create files on the mounted NFS v4 directory.

If the mount operation does not succeed, fix your installation with the help of your operating system documentation.

Configuring the NFS tiebreaker

Define a network tiebreaker as an `IBM.TieBreaker` resource of type EXEC.

The NFS tiebreaker executable `samtb_nfs` is in the `/usr/sbin/rsct/bin` directory. In the current implementation, the following options must be specified as `key=value` pairs during the creation of the RSCT exec tiebreaker:

`nfsQuorumServer`
The host name of the used NFS v4 server. This option is required.

`localQuorumDirectory`
The directory that is used by the NFS tiebreaker on the System Automation for Multiplatforms node. The directory is created automatically if it does not exist. If this option is not specified, the default directory `/var/ct/nfsTieBreaker/` is used.

`remoteQuorumDirectory`
The directory, which is exported by the `nfsQuorumServer` and used by the System Automation for Multiplatforms NFS tiebreaker. If this option is not specified, the default `/` is used.

`nfsOptions`
The options that are used for the mount command. Use the default option as documented in "Default NFS mount options" on page 61.

It is required to replace all `'='` characters by `'::'` and all `','` characters by `'..'`. For example, `vers::4..fg..soft..retry::1..timeo::10` is transformed to `vers=4,fg,soft,retry=1,timeo=10` before the mount option is passed to the mount command of the operating system.

If `nfsOptions` is not specified, the default mount options are:

AIX
`vers::4..fg..soft..retry::1..timeo::10`

Linux
`rw..soft..intr..noac..fg..retry::0`

`reserveFileName`
The file name that is created by the NFS tiebreaker in the `remoteQuorumDirectory` of the `nfsQuorumServer` to store tie breaker-related information. This option is required.

If multiple clusters are using the same NFS v4 server for the NFS tiebreaker, make sure that every cluster is using a distinct `reserveFileName`. If two clusters are using the same reserve file, one subcluster can unnecessarily lose quorum in a cluster split situation. To make sure that reserve file names are unique, you can consider a naming schema, which uses the cluster name and at least some of the cluster node names.

`Log`
Used to enable or disable writing log information to `syslog`.

- `Log=0`: No log information is gathered.
- `Log=1`: Important information is written to the `syslog`.
- `Log=2`: Trace and debug level information is produced

The default value is `1`.

`HeartbeatPeriod`
After there was a successful reserve, `ConfigRM` starts calling periodically the tiebreaker heartbeat operation. For the NFS tiebreaker, specify a value greater than `15`.

`PostReserveWaitTime`
The `PostReserveWaitTime` defines the delay between the successful reservation of the tiebreaker and the time quorum is granted. A node that reserves the tie breaker does not get operational quorum until the `PostReserveWaitTime` is passed. For the NFS tiebreaker, `PostReserveWaitTime` must be equal to `15`.

To create an NFS tiebreaker myNFSTiebreaker on NFS server my.nfs.server.com with localQuorumDirectory /my/quorumServer, log level 2, and defaults for the other options, the following command can be used:

```
mkrsrc IBM.TieBreaker Type="EXEC" Name="myNFSTie breaker"
DeviceInfo='PATHNAME=/usr/sbin/rsct/bin/samtb_nfs
nfsQuorumServer="my.nfs.server.com" reserveFileName=<unique_file_name>
localQuorumDirectory "/my/quorumServer" Log=2'
HeartbeatPeriod=30 PostReserveWaitTime=15'
```

When the NFS tiebreaker is enabled, a validation logic makes sure that the NFS server works as expected.

The following setup errors cannot be detected by the validation logic:

- If the HeartbeatPeriod is less than 15.
- If the reserveFileName is not unique for the used NFS v4 server.
- If PostReserveWaitTime is not equal to 15.

For more information about an EXEC tiebreaker, see the RSCT documentation.

Default NFS mount options

The following mount options are used:

rw
Specifies that the mounted directory is read and write accessible.

soft
Returns an error in case the NFS server cannot be reached.

intr
Allows interrupt signals.

noac
File attributes are not cached. Forces the client write request to be synchronous.

fg
Executes the mount command and fails if the mount command is not successful.

retry=0
The system gives up immediately if a mount command fails.

If you use other mount options, it cannot be guaranteed in all cases that the NFS tiebreaker still works.

Timeout protection for NFS tiebreaker operations

It is important to make sure that EXEC tiebreaker operations do not hang for the following reasons:

- RSCT-based operations like **lsrsrc**, **lsrpnode**, and **lssam** are blocked while tiebreaker operations are run.
- If a reserve operation on a node with a running critical resource hangs, the node remains in PENDING_QUORUM while another node might be able to reach HAS_QUORUM. Then, a critical resource is online on several nodes in the cluster concurrently.

The NFS tiebreaker has two processes that are defined. A worker process and a second process, which activates the timer and stops the worker if it does not finish within the timeout period:

- samtb_nfs_worker: Runs the actual tiebreaker operations.
- samtb_nfs: Initializes a timer and then runs samtb_nfs_worker from a forked thread. If samtb_nfs_worker ends within the timeout period, samtb_nfs exits with the samtb_nfs_worker return code. If samtb_nfs_worker does not end within the timeout period, the alarm handler makes sure samtb_nfs_worker is stopped, and writes an error message to syslog and ends with -1 (FAILED).

The following timeout values are used:

Reserve operation
13 seconds after the cluster split.

Validate operation
60 seconds at tiebreaker definition time.

Initialize operation
20 seconds after a node reboots during cluster initialization.

All other operations
15 seconds.

Cloud tiebreaker

The cloud tiebreaker solution resolves tie situations of reserved container, which are stored on the Amazon Web Services (S3). The cloud tiebreaker supports only two-node cluster, and AWS storage type of container.

Setting up cloud tiebreaker

The cloud tiebreaker solution resolves tie situations of TSAMP Cluster. The cloud tiebreaker supports only two-node cluster, and AWS storage type of container.

Cloud tiebreaker uses off-site cloud storage to retain tie-breaker state and provides several advantages such as avoiding split brain situations, ease of cloud usage, virtualization friendliness of network.

The cloud tiebreaker is specified by a pair of access keys and secret keys, which are used to access the cloud storage. The cloud tiebreaker service must be accessible from each node in the cluster.

The cloud tiebreaker storage service consists of containers and objects contained within these containers.

The container names must be unique because the container namespace is shared by all users of the storage service. The name of a container cannot be assigned to another container until the existing container is deleted. The containers have access control lists, which are used to authenticate connection with cloud from the node. The property of container uniqueness within the cloud service ensures that only one node of the two-node cluster can acquire the tiebreaker device. This prevents possibilities of developing split-brain situations.

This represents a cloud Tie-Breaker setup with two nodes (nodeha01 and nodeha02) in a cluster, which have access to a shared storage in the Amazon Web Services(S3). Both the nodes have read and write permissions on the cloud, so that each node can create a container on the storage and can create a tiebreaker object in the container. In case of a split brain situation, the node which have ownership of the container will have QUORUM capabilities, and the node will continue to work in the cluster.

Configuring the cloud tiebreaker

At first, define a cloud tiebreaker as an `IBM.TieBreaker` resource of the type EXEC. For more information about an EXEC tiebreaker, see the RSCT documentation. You can find the tiebreaker setup file `samtb_cld` in the `/usr/sbin/rsct/bin` directory. The script in the setup file creates a container at the remote location i.e. Amazon Web Services(S3). It also helps you delete the container and maintain ownership of the container. The node, which owns the container, will have quorum and work as the Active member of the cluster during an split brain situation .

To set up a cloud tiebreaker, complete the following procedures.

1. “Creating AWS accounts” on page 63
2. “Fetching access key and secret key from AWS” on page 63
3. “Setting up the cluster tiebreaker to the cloud” on page 64

Creating AWS accounts

Create two cloud storage accounts. The accounts must have permission to create and delete containers. You can sign up for Amazon web services (AWS) Simple Storage Service (S3).

To create cloud storage accounts on the AWS S3, complete the following steps.

1. Click the following link:
   Amazon web services (AWS) Simple Storage Service (S3).
   The browser redirects to AWS homepage.
2. Click on the `Create an AWS Account` button.
3. Enter personal details in the form displayed to create an account. And click the `Continue` button.
4. Enter payment gateway details.
   After payment gateway details is validated, your account will become active.

Fetching access key and secret key from AWS

Each node uses a distinct AWS account to access the shared cloud storage. Retrieve access key and secret key of both accounts from the website of cloud storage service. Place the access key information on each machine.

To fetch access key and secret key from AWS, complete the following steps.

1. Login to the AWS console.
2. In the home page, click your `account name`, then click `My Security Credentials`.
3. Click on the `Create New Access Key` button. Once you click the button, the browser prompts you to download the access key and secret key.

Placement of config file and keys for Python script

The efix provides a new utility `cfgsamcldtb` to configure the custom region name and `s3` bucket name. This command must be run once after the domain is online, before configuring the AWS cloud tiebreaker. In case if the region and `s3` bucket names are not configured, then default region and `s3` buckets will be used.

```
#cfgsamcldtb <region_name> <s3 bucket name>
```

This command enables user to configure the region and bucket name by creating `/var/ct/cfg/samtb_cld.cfg`:

```
region = ap-south-1
bucket_name = ctbtest-apsouth1
```

If the user does not want to configure with any of the specific region and bucket name, then the following are the default values for AWS cloud tiebreaker

```
region = us-west-1
bucket_name = <region>-cloud-tiebreaker-bucket
```

Placing key file on on-prem nodes:

Each account is associated with a pair of access key and secret key. The pair of keys must be placed in the nodes in which tiebreaker is to be setup. The access and secret keys must be placed in the files accessible to root in each of the two nodes.

Key file is referred as `samtb_cld.cred` which needs to be created under `/var/ct/cfg/`

Following is the sample content of `/var/ct/cfg/samtb_cld.cred`:

```
aws_access_key_id = AKXXXXXXXXXXXXXXXXLA
aws_secret_access_key = nJXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXjM
```

Environment validation

With Python, and the access key and secret key installed on each node, you can validate the cloud tiebreaker configuration. Run the following command on the first node with root privileges :

```
/usr/sbin/rsct/bin/samtb_cld initialize
```

Any errors indicate that prerequisites are missing. Correct any errors, then rerun the above validation command. You must not proceed further until validation is done without error.

Similarly, validate the other node.

Setting up the cluster tiebreaker to the cloud

Once you have validated that the cloud tiebreaker resource and ensured that it is correctly configured at each node of the two nodes in the cluster, run the following sequence of three commands on either node with root privileges.

Note: You should run these commands only once at either node of the cluster.

Run the following command:

```
export CT_MANAGEMENT_SCOPE=2
```

Run the following command with root privileges to create a tiebreaker resource, and to name the object `CloudTB1`:

```
mkrsrc IBM.TieBreaker Type=EXEC Name=CloudTB1 DeviceInfo=PATHNAME=/usr/sbin/
rsct/bin/samtb_cld
```

Run the following command to set the active tiebreaker for the current cluster. This command sets the newly created tiebreaker object named `CloudTB1` as the active tiebreaker:

```
chrsrc -c IBM.PeerNode OpQuorumTieBreaker=CloudTB1
```

Ensure that the three commands run without an error. After running the above commands, the two-node cluster has a 'cloud' type tiebreaker. Run the following command to validate the tiebreaker setup:

```
lsrsrc -c IBM.PeerNode OpQuorumTieBreaker
```

The output should be similar to the following screen:

```
Resource Class Persistent Attributes for IBM.PeerNode
resource 1:
        OpQuorumTieBreaker = "CloudTB1"
```

This output indicates that the newly created tiebreaker `CloudTB1` is active one in the cluster.

Problem determination and analysis

The following example shows a sample system log content for a cloud tie breaker error scenario in a two-node cluster with the logs from `nodeha01`.

The cloud tiebreaker logs into the defined native SYSLOG facility entries affixed with the following label: `1 samtb_cld`

For example, if the SYSLOG facility is storing data to the file `/var/log/messages` in this machine, you can see all entries logged by the cloud tiebreaker by running the following command:

```
cat /var/log/messages | grep samtb_cld
```

Entries of most interest are those which indicate that the quorum has been achieved. You should see messages similar to the following screen in the SYSLOG, specifically in cases where the cloud tiebreaker is able to acquire the quorum device:

```
Feb 19 15:59:03 nodeha01 samtb_cld[7203]:
*****INFO: tryReserve: returning 0
Feb 19 15:59:03 nodeha01 samtb_cld[7203]:
*****INFO: op=reserve rc=0 log=1
Feb 19 15:59:03 nodeha01 samtb_cld[7203]:
*****INFO: Exiting samtb_cld main code returning 0
Feb 19 15:59:03 nodeha01 ConfigRM[5642]: (Recorded using
libct_ffdc.a cv 2):::Error ID: :::Reference ID:  :::Template ID:
  0:::Details File:  :::Location:RSCT,PeerDomain.C,1.99.22.61,18346
  :::CONFIGRM_HASQUORUM_ST The operational quorum state
of the active peer domain has changed to HAS_QUORUM.
In this state, cluster resources may be recovered and
controlled as needed by management applications
```

Overriding the operational quorum

Override the operational quorum state if there are not enough nodes to ever achieve an operational quorum.

To remove nodes from the cluster, at least one node of the cluster must be online to initiate the **rmrpnode** command. Operational quorum is required to run this command. If there are not enough nodes to achieve operational quorum, you cannot adjust the cluster size to reestablish the quorum.

If for any reasons the operational quorum function must be deactivated, the persistent attribute `OpQuorumOverride` must be set to `1`:

```
chrsrc -c IBM.PeerNode OpQuorumOverride=1
```

In this case operational quorum State is always `HAS_QUORUM` and resource protection is not ensured anymore.

Configuring the end-to-end automation adapter

If you want to integrate a System Automation for Multiplatforms domain into the System Automation Application Manager end-to-end automation environment, you must configure the automation adapter.

To integrate a System Automation for Multiplatforms domain into the System Automation Application Manager end-to-end automation environment, the following conditions apply:

- System Automation for Multiplatforms object names. For example, group names, resource names, and descriptions), must not contain the following characters:
  - " : Double quotation mark
  - ' : Single quotation mark
  - ; : Semicolon
  - $: Dollar sign
  - / : Slash
- System Automation for Multiplatforms domain names must be unique within the scope of automation domains that connect to the same end-to-end automation manager.

Figure 13 on page 66 shows the environment in which the end-to-end automation adapter operates and what needs to be configured for the end-to-end automation adapter:

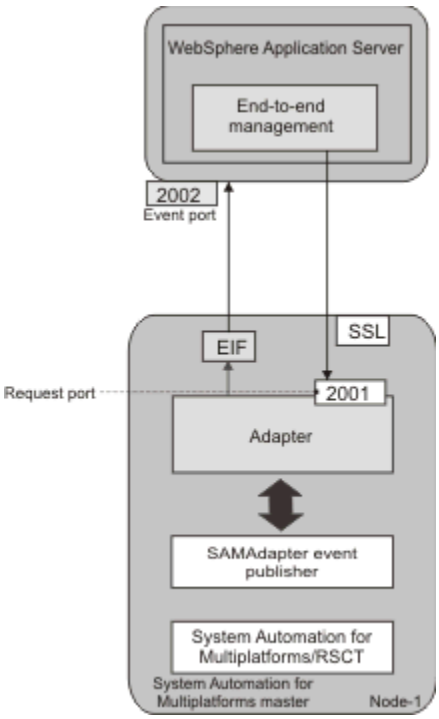

Figure 13. Overview of the environment of the end-to-end automation adapter in a System Automation for Multiplatforms cluster

To integrate a System Automation for Multiplatforms domain into the System Automation Application Manager end-to-end automation environment, the System Automation Application Manager product must be installed. For more information about end-to-end automation management, see *System Automation for Multiplatforms Administrator's and User's Guide*.

Starting the end-to-end automation adapter configuration dialog

Use the `cfgsamadapter` command to start the configuration dialog.

Note:

1. The `cfgsamadapter` utility is an X Window System application and must be used from a workstation with X Window System server capabilities. You need the X11 installation packages to run the configuration dialog. On some operating systems, those packages are contained on the distribution media, but are not part of the standard installation.
   - Install the 32-bit version of the X11 installation packages on AIX and Linux operating systems, where the 32-bit version of System Automation for Multiplatforms is installed.
   - Install the 64-bit version of the X11 installation packages on Ubuntu and Linux operating systems, where the 64-bit version of System Automation for Multiplatforms is installed.
2. On AIX systems, the following requirement must be met for the end-to-end automation adapter installation: **SSL/SSH** packages must be installed and the **sshd** subsystem must be running to be able to complete the **Replication** task of the adapter configuration.
3. You can also configure the end-to-end automation adapter in silent mode by using an input properties file. If an X11 server is not available, silent configuration is the only supported method on this system. For more information, see “Configuring in silent mode” on page 74.
4. To use the configuration dialog, log on to the system with the user ID `root` or you must have write access to the directories `/etc/opt/IBM/tsamp/sam/cfg` and `/etc/Tivoli`.

Enter the **`cfgsamadapter`** command to start the Tivoli System Automation adapter configuration dialog. The main window of the dialog is displayed:

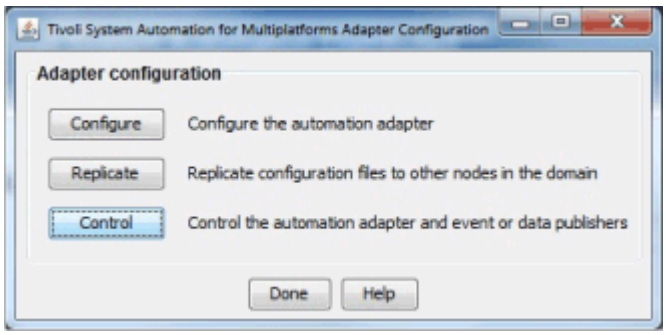

Figure 14. Main window of the end-to-end automation adapter configuration dialog

Configuration tasks:

1. Configure the end-to-end automation adapter (see page "Configuring the automation adapter settings" on page 67)
2. Replicate the end-to-end automation adapter configuration files to other nodes (see page "Replicating the end-to-end automation adapter configuration files" on page 73)
3. Control the automation adapter and event or data publishers. Start or stop the end-to-end automation adapter, the Tivoli Netcool/OMNIbus event publisher, or the reporting data publisher. For more information about the adapter and publishers, see System Automation for Multiplatforms Administrator's and User's Guide.

Configuring the automation adapter settings

On the main window of the configuration dialog, click **Configure** to display the configuration tabs that are described in the following sections.

Adapter tab

Use the **Adapter** tab to configure the adapter host.

Fields and controls on the **Adapter** tab:

Host name or IP address
Host name of the node where the adapter runs. The local host name is used as default value. If you want to use a different value than the local host name, clear the **Use local host name** check box to enable the entry field for editing. For example if you are using a second network.

Impact on configuration file replication: If you use the local host name, the **Replicate** function makes sure that the respective local host name is used on each replication target node. If you specify a different host name or IP address, the **Replicate** function replicates this value to the other nodes in the cluster. In this case configure the adapter host on each node separately, if you do not want the same value to be used on all nodes. For more information, see "Replicating the end-to-end automation adapter configuration files" on page 73.

Request port number
Specify the number of the port on which the adapter listens for requests from the end-to-end automation management host. The default port is 2001.

Policy pool location
Specify the qualified path name of the directory that contains the XML policy files. If you are using System Automation Application Manager to activate a System Automation for Multiplatforms automation policy, the policy pool is required. Define and create the policy pool directory on all nodes in the cluster. This parameter is optional.

Click **Advanced** to specify the adapter runtime behavior:

Adapter stop delay
Define the time period that is measured in seconds. The adapter stop is delayed within this time period to allow the adapter to properly deliver the domain leave event. The default value is 5. You can increase the value on slow systems. The value ranges between 3 through 60 seconds.

Remote contact activity interval
Define the time period that is measured in seconds after which the adapter stops if it was not contacted by the end-to-end automation management host. The host periodically contacts the adapter to check whether it is still running. The default value is 360. If a value other than 0 is specified, the interval must be a multiple of the check interval.

When the value is set to 0, the adapter continuously runs and never stops.

Initial contact retry interval
Define the time period that is measured in minutes. The adapter attempts within this time period to contact the end-to-end automation management host until it succeeds or the specified time elapsed. The default value is 0, which means that the adapter attempts to contact the end-to-end automation management host indefinitely.

Enable EIF event caching
Select this check box to activate event caching.

EIF reconnect attempt interval
Define the time period that is measured in seconds. The adapter will wait before it attempts to reestablish the connection to the end-to-end automation management host after the connection was interrupted. The default value is 30.

Host Using Adapter tab

Use the Host Using Adapter tab to configure the end-to-end automation manager host the adapter connects to.

Fields on the Host using adapter tab:

Host name or IP address
The name or IP address of the host on which the end-to-end automation manager runs.

Alternate host
A value for this field is optional. If you have configured a disaster recovery setup with two different sites for the System Automation Application Manager, the end-to-end automation manager may run on either site. To support such a setup, also specify the host name or IP address of the second site. In case of an Application Manager site switch, this will ensure that the adapter switches seamlessly to the new active end-to-end automation manager instance as the target for sending events.

Event port number
The port to which the end-to-end automation manager listens for events from the automation adapter. The port number specified here must match the port number specified as event port number when configuring the domain of the end-to-end automation manager. The default port is 2002.

Note: If the communication between the end-to-end automation adapter and the end-to-end automation management host uses IPv6, then the following restrictions apply.

For the communication from the adapter to the host using the adapter:

1. If an IPv6 host name is specified in the configuration of the end-to-end automation management host, the DNS server must be configured to return IPv6 records only.
2. If the DNS server is configured to return IPv4 and IPv6 records, only the IPv4 address is used. In case you want to use IPv6, explicitly specify the IPv6 address instead of the host name in the configuration of the end-to-end automation management host .

For the communication from the end-to-end automation management host to the adapter:

1. If an IPv6 host name is specified in the configuration of the adapter host, the DNS server must be configured to return IPv6 records only.

2. If the DNS server is configured to return IPv4 and IPv6 records, only the IPv4 address is used. In case you want to use IPv6, explicitly specify the IPv6 address instead of the host name in the configuration of the adapter host.

Use the command `host -n -a <ipv6_hostname>` to check the DNS lookup records.

Reporting tab

Use the Reporting tab to configure the settings to collect reporting data in the System Automation Application Manager database.

After you configure the reporting database, you need to start the reporting data publisher.

Note:

1. The reporting function, like report generation, is provided as part of the System Automation Application Manager product up to version 3.2.2.
2. Make sure to disable reporting before you uninstall System Automation Application Manager from the end-to-end automation management host.

   Local database installations of System Automation Application Manager are dropped during uninstallation. In this case, stop the reporting data publisher.

For starting or stopping the reporting data publisher, see *System Automation for Multiplatforms Administrator's and User's Guide* or use the following commands:

```
samctrl -e JDBC or samctrl -d JDBC
```

If you want to collect reporting data in the DB2® database of System Automation Application Manager, select the **Enable report data collection** check box. Otherwise, deselect the check box, which disables the entry fields on this tab.

Fields on the **Reporting** tab:

DB2 server name or IP address
The host name or IP address of the DB2 server that hosts the reporting data database. The actual reporting function, like report generation, is provided as part of the System Automation Application Manager product. The DB2 server must be the same system where the DB2 database of System Automation Application Manager is located.

If you omit this value, the value that you specify for the System Automation Application Manager host in the **Host Using Adapter** tab is used as the default. If you are using a remote DB2 for the System Automation Application Manager database, specify the host name or IP address of that remote DB2 system.

Note: If the DB2 server runs on z/OS, make sure that the file `db2jcc_license_cisuz.jar` is available on each node in your System Automation for Multiplatforms cluster. This file contains the license to connect to DB2 on z/OS from non-z/OS system.

You can find this file in the WebSphere Application Server directory that is used for System Automation Application Manager. Search the following directory tree for the file:

```
<WAS_INSTALL_ROOT>/deploytool/itp/plugins
```

Copy the file into the directory `/opt/IBM/tsamp/sam/lib` on each node in your System Automation for Multiplatformscluster. Make sure that you have a DB2 license agreement.

Alternate DB2 server
A value for this field is optional. If you configured a disaster recovery setup with two different sites for the System Automation Application Manager, the end-to-end automation manager can run on either site. To support such a setup, specify the host name or IP address of System Automation Application Manager on the second site in this field. In case of an Application Manager site switch, the adapter automatically switches to the new active end-to-end automation manager instance as the target for collected report data. The values of all following settings are used for both DB2 servers. If

the database is on the same system as the end-to-end automation manager, specify the same value that you used for the alternate System Automation Application Manager host that is using the adapter.

If you are using a remote DB2 for the System Automation Application Manager database, leave this field empty.

Note: If you specify an alternate DB2 server, you need to configure the DB2 Automatic Client Reroute feature. Then, the reporting function is enabled to always feed the reporting data to the DB2 HADR primary instance. Refer to the DB2 documentation for a description of how to set up this feature.

Example:

DB2 HADR is set up for database `eautodb` on the two hosts `lnxcm5x` and `lnxcm6x`. The DB2 port is 50000 on both hosts. To configure Automatic Client Reroute for the two hosts, run the following commands:

- On lnxcm5x:

```
db2 update alternate server for database eautodb using host name
lnxcm6x port 50001
```

- On lnxcm6x:

```
db2 update alternate server for database eautodb using host name
lnxcm5x port 50001
```

DB2 database name
The name of the DB2 database of System Automation Application Manager, where reporting data is stored.

DB2 schema name
The name of the schema that is used for the database tables where reporting data is stored. Change the value of this parameter only if the DB2 database of System Automation Application Manager is on a z/OS system. You might need to control the schema name to uniquely identify database tables in your DB2 installation.

DB2 port
The number of the port that is used to access the DB2 database of System Automation Application Manager, where reporting data is stored. The default port is 50001.

User ID
The user ID that is used to access the DB2 database of System Automation Application Manager, where reporting data is stored.

Password
The password that is used to access the DB2 database of System Automation Application Manager, where reporting data is stored.

Click **Change** to change the password.

Note: Ensure that you update the configured password whenever the DB2 database password is changed. If the configured password does not match the DB2 database password, events are not written to the database.

Event Publishing tab

Use the **Event Publishing** tab to configure settings to publish EIF events to Tivoli Netcool/Omnibus.

Controls and fields on the **Event Publishing** tab:

OMNIbus event publishing

Enable OMNIbus EIF event publishing
Select this check box if you want EIF events to be sent to the host where the OMNIbus Probe for Tivoli EIF is running. If the check box is not selected, all other fields on this tab are disabled. If you enable or disable EIF event publishing, make sure to start or stop the corresponding event publisher.

For starting or stopping the EIF event publisher, refer to *System Automation for Multiplatforms Administrator's and User's Guide* or use the following commands:

```
samctrl -e TEC or samctrl -d TECs
```

Note: For compatibility reasons, alternatively a Tivoli Enterprise Console server and port can still be configured.

Event server

Host name or IP address
The host name or IP address of the host where the OMNIbus Probe for Tivoli EIF is running. You can specify up to eight values, which are separated by commas. The first location is the primary event server, while others are secondary servers to be used in the order specified when the primary server is down.

Port number
The port number that is used by the OMNIbus Probe for Tivoli EIF to listen to EIF events. If you use port mapping, you can specify `0` as port number.

Event filter

Publish EIF events that are caused by:

Configuration changes for relationships
Select this check box if you want all EIF events that are caused by adding, removing, and changing relationships to be sent to the event server. Otherwise, configuration change events for relationships are filtered out.

Configuration changes for resources
Select this check box if you want all EIF events that are caused by adding, removing, and changing resources to be sent to the event server. Otherwise, configuration change events for resources are filtered out.

Adding and removing requests
Select this check box if you want EIF events that are caused by adding and removing requests to be sent to the event server. Otherwise, events for adding and removing requests are filtered out.

Resource status changes
Select this check box if you want EIF events that are related to resource status changes to be sent to the event server. Otherwise, all resource status change events are filtered out. Depending on the severity, select one of the radio buttons to define which status change events are published.

Defining extra filters:

The event filters that you can enable or disable on this tab are the predefined filters that are included with System Automation for Multiplatforms. If you want to define extra filters, modify manually the corresponding configuration properties file:

```
/etc/Tivoli/TECPublisher.conf
```

If you want to edit a predefined filter, add a filter and disable the predefined filter. If configuration changes are applied by the `cfgsamadapter` configuration utility, any filters that you added are preserved.

Security tab

Use the **Security** tab to configure the security for the interface between the host using the adapter and the end-to-end management host.

Select **Enable SSL** if you want to use the Secure Socket layer (SSL) protocol for the communication between the automation adapter and the host using the adapter. If checked, the following entry fields must be completed.

Controls and fields on the Security tab:

Truststore
Name of the truststore file that is used for SSL. The file name can contain multiple period characters. Click **Browse** to select a file.

Keystore
Name of the keystore file that is used for SSL. The file name can contain multiple period characters. Click **Browse** to select a file.

Keystore password
Password of the keystore file. Click **Change** to change the password.

Note: If the truststore is in different file than keystore, the passwords for the files must be identical.

Certificate alias
Alias name of the certificate to be used by the server.

Enforce user authentication
Select the **Enforce user authentication** check box to enable the authentication of the user with `Pluggable Access Module (PAM)`.

If you use System Automation Application Manager to maintain also System Automation for Multiplatforms XML policies, it is required to enable **Enforce user authentication.**

PAM Service
Name of the Pluggable Access Module service that determines which checks are made to validate users, depending on the operating system on which the adapter is running.

- For any SUSE Linux distribution, a file in directory `/etc/pam.d`
- For any RedHat Linux distribution, an entry in file `/etc/pam.conf`
- For AIX, an entry in file `/etc/pam.conf`

Logger tab

Use the Logger tab to specify the settings for logging, tracing, and First Failure Data Capture. You can change the settings permanently or temporarily.

The Logger tab always displays the values that are currently set in the configuration file.

On the Logger tab, you can perform the following tasks:

Change the settings permanently
Perform these steps:

1. Make the required changes on the tab.
2. Click **Save**.

Results: The settings in the configuration file are updated. Restart the adapter for the changes to take effect.

Change the settings temporarily
Perform these steps after ensuring that the adapter is running:

1. Make the required changes on the tab.
2. Click **Apply**.

Results: The new settings take effect immediately. They are not stored in the configuration file. If the adapter is not running, you receive an error message.

Revert to the permanent settings
If you changed the settings temporarily, perform the following steps to revert to the permanent settings defined in the configuration file, or when you are unsure which settings are currently active for the adapter:

1. Invoke the configuration dialog and open the Logger tab. The Logger tab displays the values that are currently set in the configuration file.

2. Click **Apply** to activate the settings.

   Results: The settings take effect immediately. If the adapter is not running, you receive an error message.

Controls and fields on the **Logger** tab:

Maximum log/trace file size
The maximum disk usage in KB that a log file can reach. If the limit is reached, another log file is created. The maximum number of log files is two, which means that the least recent file gets overwritten after both files are filled up. The default maximum file size is 1024 KB.

Message logging level
Select the **Message logging level**, depending on the severity of messages that you want to be logged.

Trace logging level
Select the **Trace logging level,** depending on the severity of the incidents that you want to be logged.

First failure data capture (FFDC) recording level
Select the FFDC recording level, depending on the severity of the incidents for which you want FFDC data to be collected.

First failure data capture (FFDC) maximum disk space
Specify the maximum disk space in bytes used by FFDC traces, which are written to the FFDC trace directory. The default space is 10485760 bytes (10 MB).

First failure data capture (FFDC) space exceeded policy
Select one of the options:

Ignore
Issue a warning, but do not enforce the FFDC disk space limitation.

Auto-delete
Automatically delete FFDC files to enforce the FFDC disk space limitation. This is the default value of the space exceeded policy.

Suspend
Halt further FFDC actions until disk space is freed manually.

First failure data capture (FFDC) message ID filter mode
Select one of the options:

Passthru
All log events with messages that are specified in the message ID list, pass the filter and FFDC data is written. This is the default filter mode.

Block
All log events with messages that are specified in the message ID list are blocked.

First failure data capture (FFDC) message ID list
The message IDs that control for which log events FFDC data is written, depending on the filter mode. The comparison of message IDs is case-sensitive. Each message ID must occur in a new line. Wildcard characters, for example, *E for all error messages, are allowed.

Saving the configuration

Click **Save** in the configuration window to save your changes to the adapter configuration files.

If entries are missing or a value is out of range, for example a port number, an error message is displayed. After successful completion, the configuration update status window appears, showing the list of configuration files and their update status. Restart the adapter for the changes to become effective.

Replicating the end-to-end automation adapter configuration files

Replicate the end-to-end automation adapter configuration files to other nodes in the domain.

Click **Replicate** on the main window of the configuration dialog (see “Starting the end-to-end automation adapter configuration dialog ” on page 66). The **Replicate Configuration Files** window is displayed.

Distribute (replicate) the automation adapter configuration files to the remaining nodes in the RSCT peer domain:

1. Select the configuration files that you want to replicate or click **Select all** to select all configuration files in the list.
   - If (1) file `sam.adapter.ssl.properties` is among the selected files, and (2) The SSL truststore and keystore files that you configured on the **Security** tab of the adapter configuration exist on the replication source node, then these truststore and keystore files are replicated.
   - Ensure that the directory where the files are on the replication source node also exists on all target nodes.
2. Click **Select all** below the list of replication target nodes to ensure that the adapter configuration is identical on all nodes.
3. Enter the user ID and password for the target nodes you want to replicate the files to.
4. Start the replication by clicking **Replicate**.

Replication can take a while. While the files are being replicated, the **Replicate** button is indented and grayed-out. When the replication is complete, the replication status of each configuration file is displayed.

Making the end-to-end automation adapter highly available

If the Tivoli System Automation cluster consists of more than one node, the end-to-end automation adapter must be kept highly available.

Communication to the System Automation Application Manager operations console stays alive during node outages or node maintenance in the cluster.

As illustrated in “Configuring the end-to-end automation adapter” on page 65, the automation adapter is attached to the System Automation master node. The cluster infrastructure makes sure that the master is always available and therefore also the adapter is implicitly always available on the master node. No automation policy configuration is required to make the adapter highly available starting from version 4.1.0.0 of System Automation for Multiplatforms.

Configuring in silent mode

You can configure the end-to-end automation adapter by using the silent configuration.

In the configuration tool in silent mode, you can configure the end-to-end automation adapter without starting the configuration dialog. In this case, you do not need to have an X Window session available.

Configure the end-to-end automation adapter by editing configuration parameter values in an associated properties file. If you use the silent configuration mode, you do not need to have an X Window session available.

You must first start the configuration tool to generate a silent mode input properties file before you can process a configuration update. For more information, see “Configuring the end-to-end automation adapter” on page 65.

Working in silent mode

Learn more about the major tasks if you work in the silent configuration mode.

To use the configuration tool in silent mode, you need to follow these steps for each component that you want to configure:

1. Generate or locate the silent mode input properties file, see “Silent mode input properties file” on page 75.
2. Edit the parameter values in the file, see “Editing the input properties file” on page 76.
3. Start the configuration tool in silent mode to update the target configuration files, see “Starting silent configuration” on page 75.

4. If the configuration tool does not complete successfully, deal with any errors that are reported (see "Output in silent mode" on page 76) and start the configuration tool again.

For some tasks, no silent configuration support is available. If you do not want to use the configuration dialogs, you must process these tasks manually. For more information, see "Configuration tasks to be performed manually" on page 75.

Configuration tasks to be performed manually

Some configuration tasks that are invoked in dialog mode by clicking the corresponding push button in the launchpad window are not supported in silent configuration mode.

If you do not want to use the configuration dialog, you need to manually perform the following tasks:

1. **Replicate the configuration files**

   If the System Automation for Multiplatforms domain consists of more than one node, manually replicate the end-to-end automation adapter configuration files to the other nodes in the System Automation for Multiplatforms domain. Replicate the configuration files by running the configuration tool in silent mode with identical input properties files on each node in the domain.
2. **Control the automation adapter and publishers**
   - Use command `samadapter {start|stop}` to start or stop the end-to-end automation adapter.
   - Use command `samctrl {-e|-d}` TEC to start or stop the Tivoli Netcool/OMNIbus event publisher.
   - Use command `samctrl {-e|-d}` JDBC to start or stop the reporting data publisher.

Starting silent configuration

Use command **`cfgsamadapter -s`** to start silent configuration.

Start silent configuration for the end-to-end automation adapter:

- To use the System Automation adapter configuration tool in silent mode, you must have write access to the directories `/etc/opt/IBM/tsamp/sam/cfg` and `/etc/Tivoli`.
- Enter the command `cfgsamadapter -s`

For more information about the `cfgsamadapter` command, see *Tivoli System Automation for Multiplatforms Reference*.

Silent mode input properties file

Generate a silent mode input properties file from the values that are currently configured. Use the file to modify configuration settings in silent mode.

Generate the silent mode input properties files from the values that are currently defined in the corresponding target configuration files. The advantages are:

- You can generate properties files immediately after installation and before you start to customize.
- If you customize with the configuration dialog and in silent mode, you can first generate an up-to-date input file before you apply changes in silent mode
- You can easily recover from the accidental deletion of the silent mode input properties file

To generate a silent mode input properties file, use one of the following options when you start silent configuration:

-g
Generate the input properties file only if it does not exist.

-gr
Generate the input properties file and replace it if it exists.

-l *location*
The input properties file for silent configuration is in the directory that is specified with *location*. If `-l` is omitted, the input properties file is in the default directory `/etc/opt/IBM/tsamp/sam/cfg`.

Table 21. Generated input properties files

Configuration command	Silent input properties file
cfgsamadapter -s -g | -gr	`/etc/opt/IBM/tsamp/sam/cfg/silent.samadapter.properties`
cfgsamadapter -s -g | -gr -l *location*	*location*`/silent.samadapter.properties`

If you update configuration settings in silent mode, the silent properties file is used as input for the update task. If you want the configuration utility to retrieve the input file from a location other than in the `/etc/opt/IBM/tsamp/sam/cfg` directory, use the **-l** *location* option.

Editing the input properties file

Modify the values in the input properties file to change the configuration in silent mode.

The input properties files that are generated for each of the components contain configuration parameter keyword-value pairs. To make it as easy as possible to switch between modes and to minimize errors when you edit the properties file, the structure, terminology, and wording that is used in the silent mode properties file is identical to the structure, terminology, and wording of the configuration dialog.

The names of tabs, for example **Adapter**, or buttons, for example **Advanced . . .**, on the configuration dialog are used as identifiers in the properties file, for example:

```
# ============================================================================
# ... Adapter
:
# ============================================================================
# ... Advanced
```

Each field name on the configuration dialog, for example **Request port number**, is contained in the properties file. A brief description and the keyword for that field is included, for example:

```
#     ----------------------------------------------------------------------
#     ... Request port number
#         Port of the automation adapter to receive requests from the host using
#         the adapter
adapter-request-port=2001
#
```

To edit the properties file, locate the keyword that is associated with the value you want to change and overwrite the value.

If you set the value of a required keyword to blank or comment out the keyword, the value that is defined in the target configuration file remains unchanged.

Note:

1. If a keyword is specified several times, the value of the last occurrence in the file is used.
2. Each value must be specified on one single line.

Output in silent mode

Inspect the output that is generated by the configuration tool in silent mode.

Starting the configuration tool in silent mode leads to output that closely matches the output that is displayed by the configuration dialog. The following types of output might be generated:

No update
There are no configuration updates to be saved. All parameters in all target configuration files already match the specified silent input parameters. No errors were detected when you check the silent input parameters. If additional information is available or any warning conditions are detected, the information and warnings are reported. If warnings are reported, the configuration tool issues return code "1" rather than "0". You might need to observe this behavior, when you start silent configuration, for example within a shell script.

Successful completion
At least one of the target configuration files is updated and all configuration files and their update status are listed. No errors are detected when you check the silent input parameters. If additional information is available or any warning conditions are detected, the information and warnings are reported. If warnings are reported, the configuration tool issues return code "1" rather than "0". You might need to observe this behavior when you start silent configuration, for example within a shell script.

Unsuccessful completion
No target configuration file is updated. Any errors that are detected when you check the silent input parameters are reported. The configuration tool ends, when return code "2" is returned.

Silent input properties file generation
Values from the target configuration files are used to generate the input file. No target configuration file is updated.

Unrecoverable error
Error messages that indicate the reason for the error are reported. The configuration tool ends, when return code greater than "2" is returned.

Detecting network interface failures

If you are running a single-node or two-node cluster, more configuration is required to detect network interface failures.

The cluster software periodically tries to contact each network interface in the cluster. If the attempt to contact an interface fails on one node of a two node cluster, the corresponding interface on the other node is also flagged as offline. It is flagged as offline, because it does not receive a response from its peer.

To avoid this behavior, the cluster software must be configured to contact a network instance outside of the cluster. You may use the default gateway of the sub-net the interface is in.

On each node, create the following file:

```
/var/ct/cfg/netmon.cf
```

Each line of this file contains the system name or IP address of the external network instance. IP addresses can be specified in dotted decimal format.

Example of a `netmon.cf` file:

```
#This is default gateway for all interfaces in the subnet 192.168.1.0
192.168.1.1

# This is default gateway for all interfaces in the subnet 192.168.2.0
gw.de.ibm.com
```

Using virtualized Ethernet on Power Systems

The decision about the state of network adapters, is made based on whether any network traffic can be seen on the local adapter. For example whether the local or the remote adapter is broken. Network traffic is reflected by the inbound byte count of the interface.

If Virtual I/O (VIO) is involved, the test becomes unreliable since it not possible to distinguish whether inbound traffic comes from the VIO server or client. The LPAR is not able to distinguish a virtual adapter from a real adapter. To address this problem, the `netmon` library supports up to 32 targets for each local

network adapter. If you can ping any of these targets, the local adapter is considered to be up. The targets can be specified in the `netmon.cf` file with the `!REQD` keyword.

```
!REQD <owner><target>
```

- `!REQD`: String value. No leading spaces. Always at the beginning of a line.
- `<owner>`: Specifies the interface. The `<owner>` monitors the adapter and determines the status that is based on whether it can ping any of the targets that are defined in a line below the `<owner>`. The `<owner>` can be specified as a host name, IP address, or interface name. In case the host name or IP address is specified, it must refer to the start name or IP address. No service aliases are allowed. If the host name is specified, it must be resolvable to an IP address or the line is ignored. The `!ALL` keyword specifies all adapters.
- `<target>`: The IP address or host name you want the `<owner>` to ping. A host name target must be resolvable to an IP address to be used for `netmon.cf` entries.

Running on Linux on System z under z/VM

In addition to creating the `netmon.cf` file, turn off broadcast for all communication groups when you are running System Automation for Multiplatforms on Linux on System z in a z/VM environment. The RSCT heartbeat mechanism runs a broadcast ping from time to time, especially when a network interface adapter is not available. The purpose of this feature is to find out whether the network interface adapter that sends this broadcast ping is still operational. Check if other systems reply to this broadcast ping or not. This feature is not needed if the `netmon.cf` file is set up correctly. In that case, there are other well-known network interface adapters to be checked for availability. While a broadcast ping on a stand-alone system does not present a performance issue, it has a negative impact on performance if the systems are running in a z/VM environment. The performance impact occurs, because all other systems that are running under z/VM and within the same network segment (same IP network and net mask) reply to this broadcast ping request. As a result, even VM guest systems that are idle and currently paged out are loaded into the z/VM just to reply to this ping. Depending on the number of guest systems that are running under z/VM, the performance of the whole z/VM system might decrease.

To avoid a negative impact on performance, apply the following setup changes:

- Get all the communication groups of the cluster:

  ```
  # lscomg
  ```

- Turn off broadcast for all communication groups:

  ```
  # chcomg -x b <communication group> ...
  ```

  For example:

  ```
  chcomg -x b CG1
  ```

- Use the **lscomg** command again to verify that broadcast is turned off.

Enabling disk heartbeat

You can enable disk heartbeat to ensure data integrity in cluster environments.

Disk heartbeat decreases the likelihood of a cluster split because it is able to distinguish between a network failure and a node failure.

A network failure occurs if the network connection between the nodes and from one node to the shared disk fails, as shown in Figure 15 on page 79.

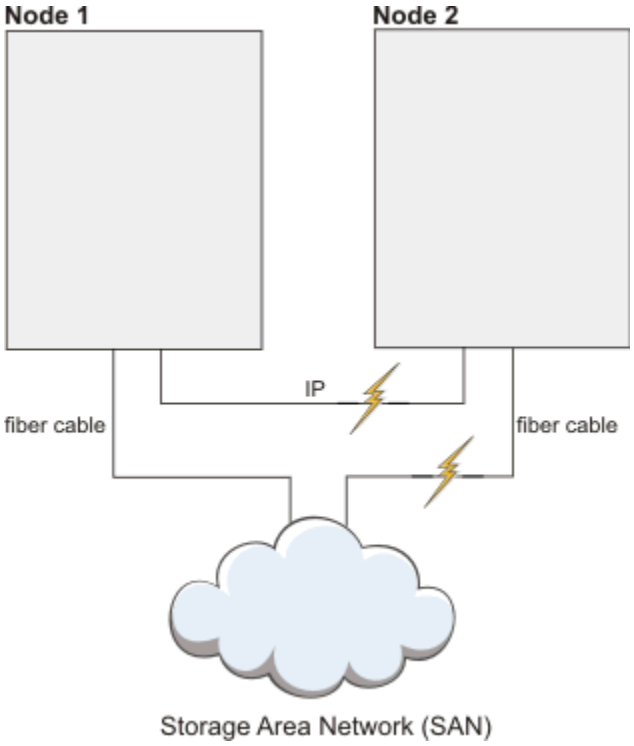

Figure 15. Network failure in a two node scenario with a shared disk

A node failure occurs if one node is not reachable any more, as shown in Figure 16 on page 79.

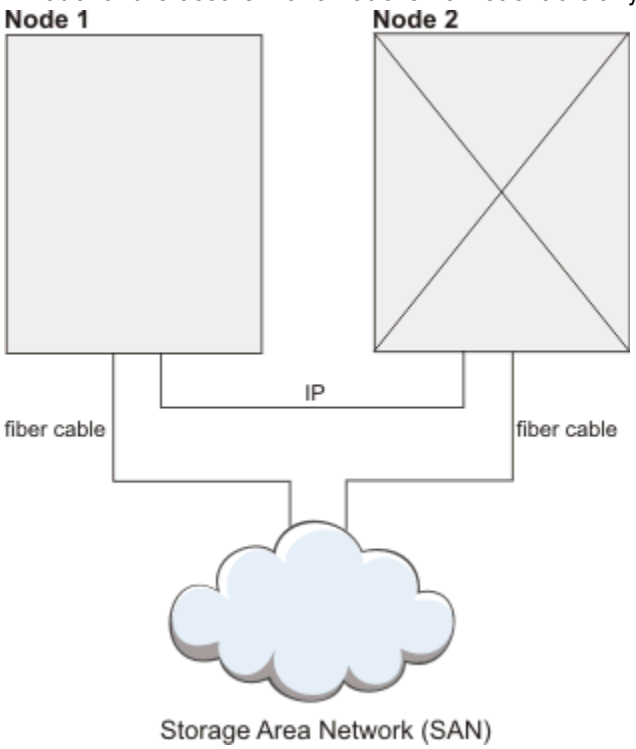

Figure 16. Node failure in a two node scenario with a shared disk

If a cluster split can get avoided, no critical resource protection is required. The systems do not need to get rebooted. Data integrity issues are avoided as well.

In case a cluster split occurs, the nodes that lost access to the heart beating disk also lose access to vital data. Critical resource protection serves to avoid data corruption. Disk heartbeat can relax critical resource protection rules since nodes without access to the disk cannot change data.

Note:

1. Disk heart beating can be enabled only when the peer domain is already online.
2. Disk heart beating can be defined only between two nodes. For more than two nodes each pair must be connected separately.

Find a suitable physical volume, logical volume, or multipath device on Linux. The data on this volume is erased. Create a heartbeat interface resource with

```
CT_MANAGEMENT_SCOPE=2
mkrsrc IBM.HeartbeatInterface attributes [Force=0|1]
```

Attributes

Name
Arbitrary name of at most 36 characters.

DeviceInfo
Valid disk or volume ID:

- `/dev/hdisk`: raw disks
- LVID: logical volumes
- MPATH: multipath devices
- PVID: physical volumes

CommGroup
Name of instance in `IBM.CommunicationGroup`. Is created if the `Force` parameter is 1.

NodeNameList
Node pair for this heartbeat interfaces, like {'node1','node2'}.

MediaType
2 (disk)

For each heartbeat ring, a communication group is created. This is also true for the conventional network-based heart beating. The communication group is created together with the heartbeat device. The communication group can be tuned similar to the network-based groups. `PingGracePeriodMilliSec` cannot be changed for disk heart beating.

Perform the following tasks to verify your disk heartbeat setup:

- In the configuration of the system setup, make sure that the disk, which is used for disk heart beating is not reserved by any peer node.
- Disk heart beating can be tested by using the following commands.

  ```
  dhb_read -p <device-name> -t    # run it on a sender side
  dhb_read -p <device-name> -r      # run it on a receiving side
  ```

  For complete verification, run the commands again, exchanging the sender and receiver node. If this test does not work, it might not be supported due to the disk reservation, or the system setup or configuration is not compatible.
- Check that the following system calls between the nodes work correctly:

  ```
  open("<dev>", O_RDWR|O_DIRECT), pread() and pwrite();
  ```

Protecting critical resources (Dead-Man-Switch)

Enable the Dead-Man-Switch (DMS) in your high availability environment.

In a high availability environment, it is crucial that at most one instance of a critical resource is running. A typical example of a critical resource is the write access to a shared disk. When write access is given to more than one node at a time, it results in utter corruption of the file system structure.

The quorum algorithms of the RSCT ConfigRM prevent this scenario from happening, if ConfigRM, HATS and HAGS are given enough system resources to run their calculations. The DMS takes effect if those RSCT infrastructure components can no longer be relied on to handle critical resources, for example due to process starvation or dead locks. The DMS requires to be accessed periodically within a certain time period. If the access fails, the operating system kernel triggers an immediate system restart to prevent a critical resource from being started twice.

On Linux systems, this function is implemented by using the `reboot` and the `halt` system call and a `softdog` module. On AIX, the device driver haDMS is used for this purpose.

Operational quorum values

If critical resources are active on a subcluster that lost quorum, the ConfigRM decides in which way the system might be stopped. Six different protection methods can be configured by the CritRsctProtMethod attribute on each node.

The following table lists which system termination methods are represented by which value of the CritRsctProtMethod attribute.

Table 22. Operational quorum protection methods

Meaning	Value
Hard reset and restart operating system (default)	1
Halt operating system	2
Hard reset and restart the operating system with `sync`	3
Halt with `sync`	4
No protection. System continues operating	5
Exit and restart RSCT subsystems	6

Enabling IPv6 Support

To use IPv6 with System Automation, you must set up your operating system for IPv4 and IPv6. Normal RSCT cluster operations use IPv4 connections, but IBM.ServiceIP resources can be defined to use IPv6 addresses.

To enable the IPv6 support in RSCT and System Automation for Multiplatforms, run the following command:

```
chrsrc -c IBM.NetworkInterface IPv6Support=1
```

The **chrsrc** command creates more IBM.NetworkInterface resources for IPv6 enabled interfaces as well. You now have two IBM.NetworkInterface resources per physical interface: one for IPv4 and one for IPv6. For examples of how to create IBM.ServiceIP resources with IPv6 addresses, see System Automation for Multiplatforms Administrator's and User's Guide. A new IBM.ServiceIP class attribute named Netprefix is defined for use with IPv6.

Setting up the automation adapter with a non-root user account

By default, the System Automation for Multiplatforms end-to-end automation adapter runs with a root user. Learn how the adapter can be set up to run with a non-root user.

Before you set up the adapter with a non-root user, configure and setup the adapter with the root user account:

- Create and start the System Automation domain.
- Configure the adapter with the `cfgsamadapter` utility.
- Configure SSL connectivity with System Automation Application Manager (optional).
- Verify the function of the adapter with the System Automation Application Manager operations console.

Processing these steps beforehand ensures that the steps for the non-root setup of the adapter must be processed one time only.

The non-root setup for the adapter comprises the following steps:

1. Perform operating system-specific security preparations, for example creation of a dedicated user and group for the adapter. Refer to “Setting up security for specific operating systems” on page 82 for a description of the corresponding actions that you must process manually.
2. Change group ownership and permissions of certain files and directories that are created by the default installation. Set the appropriate System Automation and RSCT permissions for the adapter user. The actions that are related to this step run automatically by using the script `setupAdapterNonRoot.sh`. All actions that are processed by the script are described in topic “Running the non-root user adapter setup script” on page 83.

Setting up security for specific operating systems

Learn more about operating system specific security preparations that are mandatory before you can launch the script `setupAdapterNonRoot.sh`. Perform the actions described in this section on all cluster nodes.

Creating user and group account

The same group and user account must be created on each cluster node. They are passed as input parameters to the script `setupAdapterNonRoot.sh`.

Create a group that is the primary group for the adapter user account. The group name `sagroup` is used in the following section. Any other name is valid as well. `sagroup` is used when you modify the group ownership of several files and directories of System Automation for Multiplatforms, granting access rights to the adapter user account. With System Automation for Multiplatforms version 4.1.0.4 or higher, the group can also be created by the script 'setupAdapterNonRoot.sh' when using the new option '**--manage-group**'.

Create the user account for running the adapter by using the group ID `sagroup` as the primary group for the user. The user name `samadapt` is used in the following section. The `samadapt` user account can be a technical user account that is not intended for use in a login shell. A password is not required in this case. Ensure that the home directory of the user exists and has the correct access rights.

The `samadapt` user can either be a System Automation for Multiplatforms administrator or operator. You need to follow the instructions that are provided in Chapter 5, “Securing ,” on page 111 for setting up the appropriate rights.

For an operator, assign the role `sa_operator`. For an administrator, assign the role `sa_admin`. With the `sa_operator` role, the adapter can start and stop resources and resource groups, with the `sa_admin` role it can additionally activate and deactivate policies.

Note: If you want to enable extra non-root users for System Automation administration and operation, see Chapter 5, “Securing ,” on page 111. Use the group `sagroup` for these users as well.

Configuration steps if user authentication is enabled

Extra configuration steps are required in case user authentication with Pluggable Authentication Modules (PAM) is enabled in the configuration of the automation adapter.

Linux specific (SLES)

The `samadapt` user account must be added to the shadow group ID, allowing `samadapt` to read the file `/etc/shadow`, which holds users and their encrypted passwords. The file `/etc/shadow` has the ownership of `root:shadow` with the standard permission bit settings 640. The access to `/etc/`

shadow is required to allow PAM (Pluggable Access Module) user authentication from a non-root user account. This happens when PAM is used to verify user credentials to access the System Automation for Multiplatforms domain from the System Automation Application Manager automation engine or operations console.

AIX specific

The samadapt user account must be added to the security group ID, allowing samadapt to use PAM function and access to directory /etc/security. This is required to verify user credentials when you access the System Automation for Multiplatforms domain from the System Automation Application Manager automation engine or operations console. In addition, ACL settings must be modified for the file /etc/security/password.

On AIX, the file /etc/security/passwd contains user accounts and their encrypted passwords. The file /etc/security/passwd has the ownership of root:security with the standard permission bit settings 600. This setting denies access from the samadapt user account, even if it is a member of the security group. Access can be granted by modifying ACLs on the file, avoiding modification of ownership and permission bits.

The ACLs can be modified by using the acledit or the aclget/aclput utilities. Example output:

```
*
* ACL_type   AIXC
*
attributes:
base permissions
    owner(root):  rw-
    group(security):  ---
    others:  ---
extended permissions
    enabled                      <== enable extended permissions
    permit   r--     u:samadapt  <== permit read access to samadapt
```

Merge these changes with other modifications that are applied previously.

Enable the samadapter user account to be used by System Automation Application Manager

If user authentication of the automation adapter is enabled and you want to use the samadapt user account to access the System Automation for Multiplatforms cluster from System Automation Application Manager, you need to set a password for this user ID. You can specify its credentials to access a first-level automation domain in the cfgeezdm configuration utility. Or you can use the credentials to access the domain from the System Automation Application Manager operations console.

Running the non-root user adapter setup script

Run the script setupAdapterNonRoot.sh for the remaining actions for the non-root adapter setup.

The script is in directory /opt/IBM/tsamp/sam/bin. Before you run the script, the following conditions must be met:

- If you upgrade System Automation for Multiplatforms from a version lower than 4.1 to version 4.1, all nodes in the cluster are upgraded to the new version. Rhe cluster migration is complete. The command samctrl –m is run successfully.
- The adapter is stopped.
- With System Automation for Multiplatforms version 4.1.0.3 or lower, ensure that the manual steps described in “Setting up security for specific operating systems” on page 82 completed successfully.
- The System Automation cluster is defined, but it is not required to stop the cluster. The setup steps do not interfere with cluster operations.

Run script setupAdapterNonRoot.sh on all cluster nodes.

There are different versions of this script based on the installed product version, which differ in required prerequisites and functionality. The following usage information and sample output applies to the script included within System Automation for Multiplatforms 4.1.0.0 up to version 4.1.0.3:

```
Name
setupAdapterNonRoot.sh  - configures end-to-end automation adapter to run with a non-root user account

Synopsis
      setupAdapterNonRoot.sh [-x] userName [groupName]

Description
      Script to configure the end-to-end automation adapter to run with a non-root user account.
      It adapts group ownerships and permissions, as well as RSCT security definitions.

Options
      -x  Set ACL permissions for the sa_admin role. Optional, if omitted, the default is to set ACL
          permissions for the sa_operator role.

Parameters
      userName  - the name of the user account that the adapter should run as
      groupName - the name of the primary group of the adapter user account

Exit Codes
    0 all configurations completed successfully
    1 at least one configuration task failed - see print out for details
    2 prerequisites not satisfied - see print out for details
```

Run the script as a user with root permissions:

Prerequisite checking
It is checked whether a cluster exists, the automation adapter is stopped, and the user account exists. It is also checked whether the specified group is the primary group of the user account.

Changing group ownerships and permissions
Several files and directory ownerships and permissions need to be changed, because they are initially created for root user access only. For more information, see “Changing group ownerships and permissions” on page 86.

Note: The script changes the group, which owns the file `/etc/ibm/tivoli/common/cfg/log.properties`. This file might be used by other Tivoli products as well. If one of these products is also run with a non-root user account, ensure that the `log.properties` file is still readable for these products.

Setting appropriate System Automation and RSCT permissions
To allow the non-root user account `samadapt` to use RSCT Resource Management Control (RMC), permissions must be granted by using the `/var/ct/cfg/ctrmc.acls` file. For more information, see “Setting appropriate System Automation and RSCT permissions” on page 86.

Adapting the automation adapter configuration
The non-root user and group are added to the adapter configuration properties. For more information, see “Adapting the automation adapter configuration” on page 87.

Sample output:

```
root@p6sa13 /opt/IBM/tsamp/sam/bin# ./setupAdapterNonRoot.sh -x samadapt
--------------------------------------------------------------------------------
Checking userid samadapt.
Group not set as parameter. Retrieving the primary group for user samadapt.
--------------------------------------------------------------------------------
Checking group sagroup for userid samadapt.
User account samadapt and group sagroup verified successfully. Continuing...
--------------------------------------------------------------------------------
Checking whether a Peer Domain exists  ...
Peer domain exists. Continuing ...
--------------------------------------------------------------------------------
Checking whether adapter exists and is offline ...
samadapter is not running.
Adapter exists and is offline. Continuing ...
--------------------------------------------------------------------------------
Checking for a previous non-root adapter setup ...
--------------------------------------------------------------------------------
Change various permissions. Press enter to continue ...

PolicyPool is /etc/opt/IBM/tsamp/sam/policyPool
Tivoli Common Directory is /var/ibm/tivoli/common
KeyStore not set.
TrustStore not set.
```

```
------------------------------------------------------------------------------
Replacing the DEFAULT stanza in file /var/ct/cfg/ctrmc.acls. Press enter to continue ...

Adding the following entires to the DEFAULT Stanza of /var/ct/cfg/ctrmc.acls
DEFAULT
   samadapt@0xc3d084925f78e253   *   rw
------------------------------------------------------------------------------
The command 'refresh -s ctrmc' will now be issued. Press enter to continue ...

0513-095 The request for subsystem refresh was completed successfully.
------------------------------------------------------------------------------
Adapting the file sam.adapter.properties
 Press enter to continue ...

Replacing lines in property file
------------------------------------------------------------------------------
All configurations have been completed successfully.
Run this script, including user account and group preparations on all nodes of the cluster.
If this was the last node of the cluster where you ran the script, you may now start the adapter.
```

The following usage information and sample output applies to the script included within System Automation for Multiplatforms version 4.1.0.4, and higher:

```
Synopsis:
   setupAdapterNonRoot.sh [-h] [--local] [--manage-group]
                          [-x|--sa-admin][-g|--group <groupName>]
                    userName

 Description
  Script to configure the end-to-end automation adapter to run with a non-root user account.
  It adapts group ownerships and permissions, as well as RSCT security definitions.

Parameters
  userName - the name of the user account that is used to start the adapter.

Exit Codes
  0 all configurations completed successfully
  1 at least one configuration task failed
  2 prerequisites not satisfied

Options:
  -h              Print this help.
  -g or --group <groupName> The name of the primary group for the specified user account. (default:
group name = sagroup)
  --local             Run script only on local node. Optional, if omitted, the default is to perform
changes on all cluster nodes.
  --manage-group         Create local UNIX group (if group does not exist) and add specified user to
this group.
                            Set group as primary group for the user. If omitted, the default is to not
make any changes to group and user.
  -x or –sa-admin        Set ACL permissions for the sa_admin role.Optional, if omitted, the default is
to set ACL permissions for the sa_operator role.
```

Run the script as a user with root permissions:

Prerequisite checking
It is checked whether a cluster exists, the automation adapter is stopped, and the user account exists. It is also checked whether the specified group is the primary group of the user account.

Changing group ownerships and permissions
Several files and directory ownerships and permissions need to be changed, because they are initially created for root user access only. For more information, see “Changing group ownerships and permissions” on page 86.

Note: The script changes the group, which owns the file `/etc/ibm/tivoli/common/cfg/log.properties`. This file might be used by other Tivoli products as well. If one of these products is also run with a non-root user account, ensure that the `log.properties` file is still readable for these products.

Setting appropriate System Automation and RSCT permissions
To allow the non-root user account `samadapt` to use RSCT Resource Management Control (RMC), permissions must be granted by using the `/var/ct/cfg/ctrmc.acls` file. For more information, see “Setting appropriate System Automation and RSCT permissions” on page 86.

Adapting the automation adapter configuration

The non-root user and group are added to the adapter configuration properties. For more information, see “Adapting the automation adapter configuration” on page 87.

```
Usage Examples
  1) Configure SA MP adapter to run with non-root user "saoperator" and group "sagroup"
      ("sagroup" already exists).
     Prerequisites:
     - User "saoperator" and group "sagroup" exist.
     - "sagroup" is the primary group for user "saoperator"

     Setup adapter non-root:
     # setupAdapterNonRoot.sh -g sagroup saoperator
     Result:
     - Configured SA MP adapter non-root user "saoperator" on all cluster nodes

  2) Configure SA MP adapter to run with non-root user "saoperator" and group "sagroup"
      ("sagroup" does not exist).
     Prerequisites:
     - User "saoperator" exists.
     Setup adapter non-root:
     # setupAdapterNonRoot.sh --manage-group -g sagroup saoperator
     Result:
     - Group "sagroup" is created on all cluster nodes
     - User "saoperator" is added to group "sagroup" on all cluster nodes
     - "sagroup" is set as primary group for user "saoperator" on all cluster nodes
     - Configured SA MP adapter non-root user "saoperator" on all cluster nodes
  3) Remove SA MP adapter non-root user configuration
     Prerequisites:
     - SA MP adapter non-root user is configured
     Remove adapter non-root setup
     AIX:
     # setupAdapterNonRoot.sh -g system root
     Linux:
     # setupAdapterNonRoot.sh -g root root
     Result:
     - SA MP adapter non-root user configuration is removed on all cluster nodes
```

Changing group ownerships and permissions

The script `setupAdapterNonRoot.sh` applies various changes to the group ownership of files and directories of System Automation for Multiplatforms by using the group `sagroup`. No file that owns user IDs are changed. Access permissions are also changed at the group level if needed.

Changes that are made to the file system:

- Enable the adapter to read or write its cache directory `/var/opt/IBM/tsamp`.
- Change permissions and ownership of the file `/etc/ibm/tivoli/common/cfg/log.properties`. It contains the location of the Tivoli common directory, which is used by the adapter.
- Grant read, write, or operate access to the Tivoli common directory. The directory name is stored in file `/etc/ibm/tivoli/common/cfg/log.properties`. The default directory is `/var/ibm/tivoli/common`.
- Allow reading the configuration files of the adapter in the directories `/etc/opt/IBM/tsamp/sam/cfg` and `/etc/Tivoli/tec`.
- Grant access to the adapter policy pool. The location can be configured with the `cfgsamadapter` tool. The default directory is: `/etc/opt/IBM/tsamp/sam/policyPool`.
- Change the group of the adapter binary files in `/opt/IBM/tsamp/sam/bin`, `/usr/sbin/rsct/bin`, and their related JAR files in `/opt/IBM/tsamp/sam/lib`.

For more details, see the `setupAdapterNonRoot.sh` script source.

Setting appropriate System Automation and RSCT permissions

To allow the non-root user account `samadapt` to run the RSCT Resource Management Control (RMC), the script `setupAdapterNonRoot.sh` grants permissions by using the `/var/ct/cfg/ctrmc.acls` file.

Using the script with System Automation version 4.1.0.4 or higher also adjusts the file `/var/ct/cfg/ctsec_map.global`

For more information about RSCT security, see the RSCT Technical Reference manual. The manual is packaged with the System Automation deliverable.

The `ctrmc.acls` is composed of various blocks (stanzas), which describe access permission for an RSCT resource class. In addition, the contents of the DEFAULT stanza is appended to all other stanzas. The stanza is used as a default for RSCT resource classes, that do not have its own stanza in `ctrmc.acls`. To grant access to RSCT resource classes for the adapter non-root user account, the DEFAULT stanza is modified so.

The following example of Linux SLES shows the entries that are added to the `ctrmc` DEFAULT stanza:

```
DEFAULT
root@LOCALHOST                  * rw
LOCALHOST                       * r
none:clusteruser                * r // added by preprpnode
none:root                       * rw // added by preprpnode
```

The new entries are of type `userid@RSCT-nodeid`:

userid
Non-root user account that is prepared to run the adapter.

RSCT-nodeid
RSCT `nodeid` that is contained in `/var/ct/cfg/ct_node_id` file on each cluster node.

An entry is added for each cluster node at the top of the DEFAULT stanza, so they take precedence over existing less specific entries.

You can find that the DEFAULT stanza for AIX operating systems is much larger than the Linux example. But the changes that are made are just the same.

The file ctsec_map.global is used to map local system users to RSCT users. The content is as follows:

```
unix:root@<iw>=root
unix:root@<cluster>=root
unix:*@<cluster>=clusteruser
unix:root@<any_cluster>=any_root
hba2:root@<iw>=root
hba2:root@<cluster>=root
hba2:root@<any_cluster>=any_root
```

After `ctrmc.acls` (and if applicable ctsec_map.global) modification is done, RSCT RMC is triggered to read the file again. This is done by running the command:

```
refresh -s ctrmc
```

After running the script `setupAdapterNonRoot.sh`, check the contents of `ctrmc.acls` (and if applicable ctsec_map.global) for appropriate modification.

Adapting the automation adapter configuration

When the adapter is started, it needs to know the non-root user and group.

Therefore, the script `setupAdapterNonRoot.sh` makes sure that the adapter configuration properties file `/etc/opt/IBM/tsamp/sam/cfg/sam.adapter.properties` contains the following parameters:

```
non-root-user=samadapt
non-root-group=sagroup
```

Service and Maintenance

If you install a fix pack or add nodes to the cluster, the steps for applying the non-root setup of the adapter must partly be repeated.

Scenarios and the steps that must be repeated.

Installing fix packs
Installation of System Automation for Multiplatforms fix packs might replace files and corresponding group ownerships and permissions in the `/opt/IBM/tsamp/sam` directory.

Run the script `setupAdapterNonRoot.sh` again on each node right after you installed a fix pack on the node. Specify the same input parameters for the script as with its initial invocation.

Adding new nodes
Add a node to the cluster by using the `preprpnode` and `addrpnode` command.

Run the steps that are described in "Setting up security for specific operating systems" on page 82 on the new node after you added the node to the cluster. Run the script `setupAdapterNonRoot.sh` as described in "Running the non-root user adapter setup script" on page 83 on all nodes of the cluster (old and new). Specify the same input parameters for the script as with its initial invocation on the former cluster nodes.

Changing the non-root adapter user ID

If you want to change the user ID that is used for the non-root adapter setup, remove the existing setup. Then you can define the setup for the new user.

Remove the existing setup by running the script `setupAdapterNonRoot.sh` with the following parameters. For System Automation for Multiplatforms version 4.1.0.0 – 4.1.0.3 use:

```
setupAdapterNonRoot.sh -x root
```

Then run the script again with the wanted new user ID and group.

For System Automation for Multiplatforms version 4.1.0.4 or higher use:

```
AIX:
setupAdapterNonRoot.sh -g system root
Linux:
setupAdapterNonRoot.sh -g root root
```

Then run the script again with the wanted new user ID and group.

Removing the non-root adapter setup

Remove the non-root adapter setup by resetting all permissions and authorizations to root.

Run the script `setupAdapterNonRoot.sh` with the following parameters:

For System Automation for Multiplatforms version 4.1.0.0 – 4.1.0.3 use:

```
AIX:
setupAdapterNonRoot.sh -x root system
Linux:
setupAdapterNonRoot.sh -x root root
```

For System Automation for Multiplatforms version 4.1.0.4 or higher use:

```
AIX:
setupAdapterNonRoot.sh -g system root
Linux:
setupAdapterNonRoot.sh -g root root
```

Limitations

The limitations are related to problems that might arise when you access XML policies in the System Automation for Multiplatforms policy pool. Limitations can occur when you replicate configuration files to other nodes in the cluster.

Starting the adapter with an active policy that is not readable by the non-root user account
When the policy is loaded from an XML policy file, the name and location of this file can be displayed if you enter the `lssamctrl` command from a command shell on one of the cluster nodes. The location

of the policy file does not need to be the policy pool, because the `sampolicy -a` command can use policy files from any location.

```
node:~ # lssamctrl
Displaying SAM Control information:

SAMControl:
TimeOut = 60
RetryCount = 3
Automation = Auto
ExcludedNodes = {}
ResourceRestartTimeOut = 5
ActiveVersion = [3.2.2.2,Mon Apr 8 15:49:33 2013]
EnablePublisher = Disabled
TraceLevel = 31
ActivePolicy = [/etc/opt/IBM/tsamp/sam/policyPool/nonRootAdapter-testuser-2.xml,20130415143902+0200,0]
CleanupList = {}
PublisherList = {}
```

In case this XML policy file exists but is not readable by the non-root user account. Then, the adapter fails to start properly on that node and the connection to System Automation Application Manager is not established.

Resolution: Modify the permission of the XML policy file or move the file to the policy pool.

Reading and activating System Automation for Multiplatforms policies from the policy pool. This is not possible if `samadapt` has operator role.

To activate a new or changed automation policy from the System Automation Application Manager operations console, the `samadapt` user ID must have the permission to read the corresponding XML file in the policy pool. XML policies that have inappropriate ownership or permission bit settings are not displayed in the policy selection dialogs of the operations console.

Resolution: The non-root setup steps adjust ownership and permission for existing XML policy files. Ensure that XML policy files that are stored into the policy pool later, for example by saving policies with the `sampolicy -s` command, have appropriate permissions.

Replicating configuration files

Replicate configuration files to other nodes in the cluster by using the **Replicate** function of the `cfgsamadapter` utility. Some of the replaced files have write permissions set for the root user ID only. Therefore, you can run the **Replicate** function only if you use the root user ID.

Resolution: Run the `setupAdapterNonRoot.sh sript` on the replication target nodes immediately after the replication finished. Specify the same input parameters for the script as with its initial invocation. As an alternative to using the **Replicate** function run `cfgsamadapter` to run the same configuration changes explicitly on each cluster node.

Chapter 4. Integrating

System Automation for Multiplatforms integrates other Tivoli applications to provide a comprehensive solution. The integration of Tivoli applications and your environment requires specific configuration tasks to adapt to your existing infrastructure.

The required configurations for the following integrations are described:

- Forward System Automation for Multiplatforms events to IBM Tivoli Enterprise Console® (TEC).
- Forward System Automation for Multiplatforms events to IBM Tivoli® Netcool/OMNIbus.
- Enrich TBSM views with information from System Automation for Multiplatforms resources and events.

Event consoles

System Automation for Multiplatforms sends EIF events to either Tivoli Enterprise Console (TEC) or Tivoli Netcool/OMNIbus (OMNIbus). TEC and OMNIbus are rule-based event management applications that use a central server to process incoming events.

They collect alarms and events from a variety of sources:

- Tivoli applications
- Tivoli partner applications
- Customer applications
- Network management platforms
- Relational database systems

For IBM Tivoli System Automation for Multiplatforms an event is generated and forwarded to the TEC or OMNIbus event console in the following cases:

- The configuration of IBM Tivoli System Automation for Multiplatforms or the state of an automated resource changes.
- Problems are encountered.

If you want to use System Automation events with Tivoli Business Service Manager (TBSM), you must forward the events to OMNIbus.

IBM Tivoli System Automation for Multiplatforms can produce the following types of events:

Table 23. System Automation Application Manager event class types

Event Class / Alert Group	**Description**
SystemAutomation_Resource_Status_Change	Status of an automated resource changed.
SystemAutomation_Resource_Configuration_Change	A new automated resource has been added or an existing resource has been deleted or modified.
SystemAutomation_Relationship_Configuration_Change	A new relationship has been added or an existing relationship has been deleted or modified.
SystemAutomation_Domain_Status_Change	The domain status changed. For example: • The automation manager or the automation adapter of the domain starts or stops. • A new automation policy is activated.
SystemAutomation_Request_Configuration_Change	A new request has been issued against an automated resource or an existing request has been cancelled.

The following topics describe how to set up IBM Tivoli System Automation for Multiplatforms and the event consoles to enable event forwarding to either TEC or OMNIbus:

- Set up OMNIbus to use with IBM Tivoli System Automation for Multiplatforms: "Tivoli Netcool/OMNIbus" on page 92
- Set up TEC for use with IBM Tivoli System Automation for Multiplatforms: "Tivoli Enterprise Console" on page 100.

After you prepare the event console of your choice, you must enable event generation as described in "Enabling event generation" on page 100.

Tivoli Netcool/OMNIbus

The topics in this section describe how to set up IBM Tivoli Netcool/OMNIbus to forward System Automation events to the OMNIbus event console. This OMNIbus set up is also a prerequisite for the integration of IBM Tivoli System Automation for Multiplatforms with Tivoli Business Service Manager.

Prerequisites

As System Automation for Multiplatforms uses Tivoli Event Integration Facility (EIF) events for communication, the following components are required:

- IBM Tivoli Netcool/OMNIbus (OMNIbus)
- OMNIbus Probes Library for Nonnative Base
- OMNIbus Probe for Tivoli EIF (EIF Probe). This probe can receive EIF events sent from System Automation and forward them to the ObjectServer.

The following minimum versions are required:

- OMNIbus Probe for Tivoli EIF V.9.0
- IBM Tivoli Netcool/OMNIbus 7.2.1

Note: If you are running IBM Tivoli Netcool/OMNIbus V7.2.1, install Interim Fix 3 (7.2.1.5-IF0003). If you are running IBM Tivoli Netcool/OMNIbus V7.3 or higher, no additional fix packs are required.

Install and set up these components according to the documentation available in the IBM Tivoli Netcool/OMNIbus Knowledge Center.

Environment variables

$NCHOME
Refers to the Netcool® home directory into which the packages are installed. Default directory under Linux: `/opt/IBM/tivoli/netcool`.

$OMNIHOME
The `$OMNIHOME` variable is used to provide legacy support for scripts, third-party applications, and probes that continue to use the `$OMNIHOME` environment variable. `$OMNIHOME` refers to `$NCHOME/omnibus`.

Event fields in OMNIbus database

The OMNIbus `alerts.status` table will be extended with the following new columns to hold System Automation for Multiplatforms specific information. They will be filled in the System Automation for Multiplatforms specific OMNIbus rules file when processing an event.

Table 24. System Automation for Multiplatforms status attributes used in resource status change events (alerts.status)

Attribute Name	Type	Description
SADesiredState	varchar(16)	Desired State reflecting the automation goal of an automated resource. Possible values: • Online • Offline • NoChange This means the automation goal of the resource cannot be changed by an operator
SAObservedState	varchar(16)	Current observed state of an automated resource possible values: • Unknown • Online • Offline • Starting • Stopping • NotApplicable **Note:** Corresponds to `c_status_observed` in TEC events.
SAOperationalState	varchar(255)	List of operational state values giving more fine-grained information about the current state of the resource. For a list of possible values, see the `SystemAutomation.baroc` file. **Note:** Corresponds to `c_status_operational` in TEC events.
SACompoundState	varchar(16)	Compound state indicating whether the resource is working as desired or has encountered an error. Possible values: • Ok • Warning • Error • Fatal **Note:** Corresponds to `c_status_compound` in TEC events.

Table 25. Resource, domain, event identification (alerts.status)

SADomainName	varchar(64)	Name of the Automation Domain. Part of the resource key to identify a resource. **Note:** Corresponds to `sa_domain_name` in TEC events

Table 25. Resource, domain, event identification (alerts.status) (continued)		

SAResourceName	varchar(255)	Name of the resource. This is a compound resource name consisting of the resource name itself concatenated with the resource class and optionally the resource node. The order of the name parts and the separator character depends on the sending System Automation product.
		For SA MP and SA AM:
		`<class_name>:<resource_name>:<node_name>`
		For SA z/OS:
		`<resource_name>:<class_name>:<node_name>`
		Note: <node_name> is only set if it exists. For System Automation Application Manager resource references, the node name contains the name of the referenced first level automation domain. Corresponds to `sa_resource_name` in TEC events.

SAEventReason	varchar(255)	Event reasons. One event can have multiple event reasons in TEC event. Examples for event reasons: • StatusCommonObservedChanged • ConfigurationDeleted • PreferredMemberChanged **Note:** Corresponds to `sa_event_reason` in TEC events.
SAReferencedResource	varchar(255)	For System Automation Application Manager end-to-end resource references, this contains the referenced resource key.

Table 26. Other attributes used in resource status change events (alerts.status)		
SAExludedFromAutomation	varchar(16)	Flag indicating if the resource is excluded from automation (i.e. automation is suspended). Used in resource status change events. Possible values: • NotExcluded • Excluded **Note:** Corresponds to `sa_flag_excluded` in TEC events.
SADesiredRole	varchar(16)	Desired role. Used for replication references indicating the desired storage replication direction (SA AM only). Used in resource status change events. **Note:** Corresponds to `sa_role_desired` in TEC events.
SAObservedRole	varchar(16)	Observed role. Used for replication references indicating the observed storage replication direction (SA AM only). Used in resource status change events. **Note:** Corresponds to `sa_role_observed` in TEC events.

Table 27. Domain status change events (alerts.status)		
SADomainState	varchar(16)	Status of automation domain, possible values are: • Online • Offline • Unknown **Note:** Corresponds to `sa_domain_state` in TEC events.
SACommunicationState	varchar(32)	This state reflects the connection and availability state of the domain, if the domain is connected to System Automation Application Manager. Possible values: • Ok • AsyncTimeout • AsyncMissedEvent • SyncFailed • SyncFailedAndAsyncMissedEvent • SyncFailedAndAsyncTimeout • DomainHasLeft **Note:** Corresponds to `sa_communication_state` in TEC events.

Beside the new fields for System Automation events, the following existing fields will be set in the rules file for System Automation events during event processing.

Table 28. Existing rules file fields for System Automation events	
Attribute Name	**Description**
Manager	Descriptive name of the probe that collected and forwarded the alarm to the ObjectServer. Value for SA events: `tivoli_eif on <host name>`.
Agent	Descriptive name of the manager that generated the event. Value for System Automation for Multiplatforms events: `SystemAutomation`.
Node	Identifies the host name from which the event comes from.
AlertGroup	Identifies the type of event issued by System Automation. See Table 23 on page 91 for a list of possible event classes.
AlertKey	Descriptive key that indicates the resource that triggered the event. For resource events, it contains the resource key formatted as System Automation source token, e.g. `EEZResourceKey,DN={DB2Cluster},NN={},RN={db2rs},RC={IBM.Application}`. For domain events, it contains the domain name formatted as System Automation Source Token, e.g. `EEZDomain,DN={Db2Cluster}`

Table 28. Existing rules file fields for System Automation events (continued)	
Attribute Name	**Description**
Severity	Indicates the event severity level. For resource events, the compound state of the resource determines the severity level. The color of the event in the event list is controlled by the severity value: • 0: Clear • 1: Indeterminate • 2: Warning • 3: Minor • 4: Major • 5: Critical See "Compound state to severity mapping" on page 96.
Summary	Text summary describing the event.
Service	Name of the service affected by this event. Corresponds to field `SAResourceName`.
Identifier	Identifier which uniquely identifies the problem source and controls ObjectServer deduplication. The ObjectServer uses deduplication to ensure that event information generated from the same source is not duplicated in the event list. Repeated events are identified using the Identifier attribute and stored as a single event to reduce the amount of data in the ObjectServer. For System Automation events, the Identifier field is set to `AlertKey + ":" + AlertGroup`. Therefore, the event console displays always the last event of the same resource and AlertGroup.
Class	The unique class for System Automation events. Value is 87725 (Tivoli System Automation).
ExtendedAttr	Holds name-value pairs of additional internal System Automation specific attributes, for which no dedicated column exists in the `alerts.status` table.

In addition to these attributes which are stored in the OMNIbus `alerts.status` table, extra information is written to the `alerts.details` table. For example, for domain events the product name and version of the automation product corresponding to the domain are stored in the `alerts.details` table.

Compound state to severity mapping

For events that contain a `SACompoundState` value, for example all resource state change events, the following mapping table is used:

Table 29. Compound state to OMNIbus severity mapping	
SACompoundState	**OMNIbus Severity field**
Fatal	5 (Critical)
Error	4 (Major)
Warning	3 (Minor)
OK	1 (Indeterminate)

For other events that do not contain the `SACompoundState` value, for example request events or domain events, the EIF severity field is used to determine the OMNIbus severity.

Table 30. EIF to OMNIbus severity mapping	
EIF Severity	**OMNIbus Severity field**
60 (FATAL)	5 (Critical)
50 (CRITICAL)	5 (Critical)
40 (MINOR)	4 (Major)
30 (WARNING)	3 (Minor)
20 (HARMLESS)	2 (Warning)
Else	1 (Indeterminate)

Note: The EIF Severity value of the original EIF event can be found in the `ExtendedAttr` field of an event.

Configuring OMNIbus to process System Automation events

Configuring OMNIbus involves updating the OMNIbus database and enabling the rules file.

Updating the OMNIbus database

The OMNIbus ObjectServer database includes the `alerts.status` table which contains all fields that are shown and selected by an event list.

For System Automation for Multiplatforms events, the additional columns described in "Event fields in OMNIbus database" on page 92 have to be created in the `alerts.status` table.

The `sa_db_update.sql` file creates the new columns in the `alert.status` table. The event class used for events from Tivoli System Automation is created as well. System Automation for Multiplatforms uses the event class 87725 for its events. The class is used to associate tools like the launch-in-context tool, to a specific type of event.

Enter the following command on the OMNIbus server:

UNIX:

```
$OMNIHOME/bin/nco_sql -server NCOMS -username root < sa_db_update.sql
```

Windows:

```
%NCHOME%\bin\redist\isql -S NCOMS -U root < sa_db_update.sql
```

Enter your password when prompted.

You can find the file `sa_db_update.sql` on the System Automation for Multiplatforms product DVD in the directory `/integration`.

Note: The event class 87725 is predefined in OMNIbus Version 7.3.1 or higher. If you run the `sa_db_update.sql` script using OMNIbus Version 7.3.1, you receive the following error message:

```
ERROR=Attempt to insert duplicate row on line 2 of statement 'insert into
alerts.conversions values ( 'Class87725','Class',87725,'Tivoli System
Automation' );...'
```

You can ignore this error message.

Verify that the SA specific columns and event class has been successfully added to OMNIbus:

1. Open the Netcool/OMNIbus Administrator window using the `nco_config` command.
2. From the Netcool/OMNIbus Administrator window, select the **System** menu button.
3. Click **Databases**. The Databases pane opens.
4. Select the **alerts.status** table. The `alerts.status` table pane opens.

5. Verify that the following columns are listed:
    a. SACompoundState
    b. SADesiredState
    c. SAObservedState
    d. SAOperationalState
    e. SADomainName
    f. SAResourceName
    g. SAReferencedResource
    h. SAEventReason
    i. SAExludedFromAutomation
    j. SADesiredRole
    k. SAObservedRole
    l. SADomainState
    m. SACommunicationState
6. From the Netcool/OMNIbus Administrator window, select the **Visual** menu button.
7. Click **Classes**. The Classes pane opens.
8. Verify that the class with ID **87725** and label **Tivoli System Automation** is listed in the table.

Enabling rules file

An OMNIbus rules file defines how the probe processes event data to create an alert. For each alert, the rules file also creates an identifier that uniquely identifies the problem source.

The probe for Tivoli EIF uses a standard rules file named `tivoli_eif.rules`. System Automation for Multiplatforms ships the System Automation specific rules file `tivoli_eif_sa.rules`. This file needs to be included within the default `tivoli_eif.rules` using an include statement. The rules file `tivoli_eif_sa.rules` processes an EIF event received by the probe for Tivoli EIF if the event field `source` contains the value `System Automation`.

The default `tivoli_eif.rules` file is on the system where the probe for Tivoli EIF is installed in the following directory:

```
Windows: %OMNIHOME%\probes\<os_dir>\tivoli_eif.rules
UNIX: $OMNIHOME/probes/<os_dir>/tivoli_eif.rules
```

Perform the following steps to enable the `tivoli_eif_sa.rules` file:

1. Copy the file `tivoli_eif_sa.rules`, which is in the `/integration` directory on the System Automation for Multiplatforms product CD to the system where the OMNIbus probe for Tivoli EIF is installed. As target directory, choose the directory where the `tivoli_eif.rules` file is located.
2. Enable the shipped rules file `tivoli_eif_sa.rules`. Edit the `tivoli_eif.rules` file that is used in the probe for Tivoli EIF and add an include statement for the `tivoli_eif_sa.rules` file.

   The content of the `tivoli_eif.rules` looks different depending on the type of OMNIbus installation you have:

   a. If you use a stand-alone OMNIbus installation:

      Open `tivoli_eif.rules` file in a text editor and add the include statement after the `switch($source)` block:

```
:
else
{
        switch($source)
        {
                case "dummy case statement": ### This will prevent syntax errors in case
                            no includes are added below.
```

```
                        include "tivoli_eif_tpc.rules"
                        include "tivoli_eif_tsm.rules"

                        # Uncomment the following line when using TADDM integration
                        # This rules file is available in OMNIbus 7.3 and newer only
                        # include "tivoli_eif_taddm.rules"

                default:
                        # Comment out the following line when not receiving events from TEC
                        include "tivoli_eif_default.rules"
        }
        include "tivoli_eif_sa.rules"
}
```

b. If you integrate with Tivoli Business Service Manager (TBSM) and use the OMNIbus version that is packaged with TBSM:

   Open the `tivoli_eif.rules` file in a text editor and add the include statement in the block where the predefined rules files are included. Search for the line `# Include customer rules which would override any previous rules.` and add the include statement for `tivoli_eif_sa.rules` before this line:

```
:
:
    ###
    ### Handle TEC Events
    ###
    include "tec_event.rules"

    ###
    ### Handle Z Events
    ###
    # include "zos_event.rules"

    ###
    ### Handle Z user defined events.
    ###
    # include "zos_event_user_defined.rules"

    ###
    ### Handle Z identity assignement.
    ###
    # include "zos_identity.rules"

    ###
    ### Handle EE( Event Enablement) events.
    ###
    # include "tivoli_eif_ee.rules"

   include "tivoli_eif_sa.rules"

    # Include customer rules which would override any previous rules.
    # include "customer_override.rules"
:
:
```

3. Stop the EIF probe.
   - On Windows: Select **Control Panel > Administrative Tools > Services**. In the list of services, double-click the **EIF probe**, then click **Stop**.
   - On UNIX: Enter the following command on the command line

     ```
     $OMNIHOME/bin/nco_pa_stop -process <probe_name>
     ```

4. Restart the EIF probe.
   - On Windows: In the list of services, double-click **OMNIbus EIF Prob**e, then click **Start**
   - On UNIX: Enter the following command on the command line:

     ```
     $OMNIHOME/bin/nco_pa_start -process <probe_name>
     ```

Note:

1. You can test your changes in the rules file by using the syntax checking tool `nco_p_syntax` delivered with the OMNIbus server. Use the root rules file `tivoli_eif.rules`. Included files are checked automatically.

   Example:

```
$OMNIHOME/probes/nco_p_syntax -rulesfile $OMNIHOME/probes/linux2x86/tivoli_eif.rules
```

2. If you want the Probe to be forced to read again the rules file without losing events, enter the following command:

   ```
   kill -HUP <pid>
   ```

   `pid` is the probe process ID. You can determine the `pid` by using the `nco_pa_status` command.

Tivoli Enterprise Console

You can configure the Tivoli Enterprise Console® to forward System Automation events to the TEC.

Configuring TEC to process System Automation events

The programming language Basic Recorder of Objects in C (BAROC) is used to define the structure of events and their properties. These definitions are stored in files with the extension `.baroc`. The baroc file for System Automation events is called `SystemAutomation.baroc` and is located in directory `/usr/sbin/rsct/samples/tec/SystemAutomation.baroc` after the installation. To prepare TEC to use with System Automation for Multiplatforms, import, compile, load, and activate the TEC `baroc` file `SystemAutomation.baroc` in the TEC server. For more information refer to IBM Tivoli Enterprise Console Rule Builder's Guide , GC32–0669.

Enabling event generation

If you want to send events to TEC or OMNIbus, enable event forwarding in System Automation for Multiplatforms.

Activate and configure the EIF event generation and forwarding function by enabling the TEC publisher. Perform the following steps:

1. Configure event publishing using the `cfgsamadapter` configuration utility. For more information about how to configure event publishing, refer to “Event Publishing tab” on page 70.
2. Enable the publisher on each node in the System Automation for Multiplatforms cluster. By default, the publisher is disabled. You can enable the publisher using either the configuration dialog *System Automation for Multiplatforms Administrator's and User's Guide* or by using the command `samctrl` as described in “Enabling publisher using the command line interface” on page 100.
3. Set a new language locale for the TEC event messages if you do not want to use the default system locale.

Enabling publisher using the command line interface

You can use either the System Automation for Multiplatforms command line interface (CLI) or the `cfgsamadapter` configuration dialog to control the publisher.

This section describes how to control the publisher using the CLI. If you want to use the `cfgsamadapter` configuration dialog, refer to *System Automation for Multiplatforms Administrator's and User's Guide*.

The Publisher function is disabled by default. To query the status of the publisher, issue the following command:

```
node1:/usr/sbin/rsct/samples/tec # lssamctrl
```

The following Tivoli System Automation control information is displayed:

```
SAMControl:
    TimeOut            = 60
    RetryCount        = 3
    Automation        = Auto
    ExcludedNodes        = {}
    ResourceRestartTimeOut    = 5
    ActiveVersion        = [3.2.0.0,Wed Feb 17 20:19:07 2010]
    EnablePublisher        = XDR_GDP2 XDR_GDP1
    TraceLevel        = 31
    ActivePolicy        = []
    CleanupList        = {}
    PublisherList        = {}
```

To enable the TEC publisher, issue this command on any node:

```
node1:/usr/sbin/rsct/samples/tec # samctrl –e TEC
```

To disable the TEC publisher, issue this command on any node:

```
node1:/usr/sbin/rsct/samples/tec # samctrl –d TEC
```

To enable all defined publishers, issue this command on any node:

```
node1:/usr/sbin/rsct/samples/tec # samctrl –e P
```

To disable all defined publishers, issue this command on any node:

```
node1:/usr/sbin/rsct/samples/tec # samctrl –d P
```

Setting a new language locale for the TEC or OMNIbus event messages

TEC or OMNIbus event messages are always in the language which is the default system locale on the node where the System Automation for Multiplatforms master is running.

Note: Resource names in TEC or OMNIbus event messages can be corrupted, if the user created the resources (mkrg, mkrsrc) in a shell with a different locale than the default system locale, or the terminal program has a different character set translation defined than the shell locale. To solve this problem, the system and shell locales must have identical settings and the character translation of the terminal program must be set accordingly. If the shell locale changes and resources are already created with the old shell locale setting, all resources must be deleted and have to be recreated with the new shell locale.

If the user chooses to adjust the default system locale to his preferred shell settings, then this change has to be done on all nodes of the cluster. Do the following to perform this:

1. Stop the cluster using the **stoprpdomain** command.
2. Edit the file containing the default system locale, set the appropriate values, and save the file.

   SUSE Linux

   File: `/etc/sysconfig/language`

   Keywords: `RC_LANG="<NewLocale>"`

   Replace `<NewLocale>` with your locale setting.

   `ROOT_USES_LANG="yes"`

   All keywords starting with `RC_LC_` must be set to empty strings "", for example `RC_LC_ALL= ""`.

   Run `/etc/SUSEconfig` to apply the changes to your system. You can also use the `yast2 sysconfig` system configuration tool to apply the changes.

   RedHat Linux

   File: `/etc/sysconfig/i18n`

   Keywords: `LANG="<NewLocale>"`

Replace <NewLocale> with your locale setting.

AIX

File: /etc/environment

Keywords: LANG="<NewLocale>"

Replace <NewLocale> with your locale setting.

3. Reboot the system.
4. Repeat the steps on all nodes in the cluster.
5. Start the cluster using the **startrpdomain** command.

Tivoli Business Service Manager (TBSM)

TBSM delivers the real-time information that you need in order to respond to alerts effectively and in line with business requirements, and optionally to meet service-level agreements (SLAs).

The TBSM tools enable you to build a service model that you integrate with IBM Tivoli Netcool®/ OMNIbus™ alerts or optionally with data from an SQL data source.

The TBSM Data server analyzes IBM Netcool/OMNIbus ObjectServer events or SQL data for matches against the incoming-status rules you configured for your service models. If the matching data changes the service status, the status of the TBSM service model changes accordingly. When a services status changes, TBSM sends corresponding service events back to the ObjectServer.

The Discovery Library Toolkit lets you create TBSM service objects using data from Discovery Library Adaptor (DLA) books or from the IBM Tivoli Application Dependency Discovery Manager.

The TBSM console provides a graphical user interface (GUI) running in the Tivoli Integrated Portal (TIP) that allows you to logically link services and business requirements within the service model. The service model provides an operator with a view of how an enterprise is performing at any given moment in time or how the enterprise has performed over a given time period.

The following picture shows the basic architecture for TBSM:

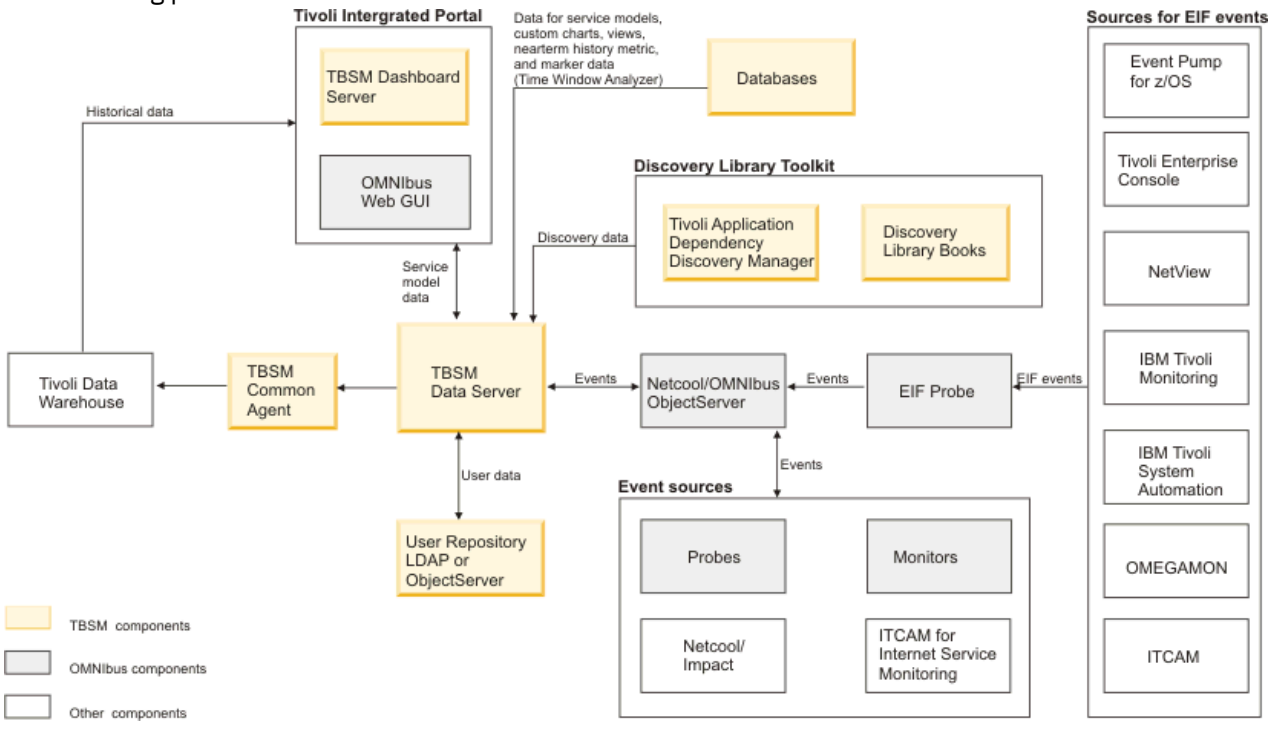

Figure 17. TBSM basic architecture

Main Components

Tivoli Integrated Portal
Tivoli Integrated Portal enables the interaction and secure passing of data between Tivoli products through a common portal. You can launch from one application to another and within the same dashboard view to research different aspects of your managed enterprise.

Tivoli Netcool/OMNIbus
TBSM monitors the Tivoli Netcool/OMNIbus ObjectServer for incoming events. The ObjectServer collects events from probes, monitors, and other applications such as IBM Tivoli Monitoring. You use TBSM to create service models that respond to the data received in the incoming events. For example, the incoming event data can change the status of a service or start the tracking of a potential SLA violation.

Tivoli Netcool/Webtop (OMNIbus Web GUI)
Netcool/Webtop is the browser console for Netcool/OMNIbus and TBSM uses Netcool/Webtop components to display events related to service models. The Active Event List (AEL) and Service Details portlet in TBSM are Netcool/Webtop components, and are installed as part of TBSM. The Tivoli Integrated Portal also includes Netcool/Webtop components.

TBSM Dashboard server
The TBSM Dashboard server manages the TBSM console display and communicates with the TBSM Data server to support the creation and visualization of service models through connected TBSM consoles. As console users view portions of the service model, the dashboard server will acquire and maintain status of services from the data server.

TBSM Data server
The TBSM Data server monitors the ObjectServer and external databases for data that affect the status of the services you configured in the TBSM console or with the radshell command line tool. The server calculates the status of these services by applying rules to the external data. Your service models and the rules are stored in the TBSM database.

Integrating System Automation for Multiplatforms

Business applications typically consist of different middleware components, are multi-tiered, and run on heterogeneous platforms. Tivoli Business Service Manager (TBSM) provides health information about the multitiered application. TBSM also monitors service level agreements (SLA) based on information coming from numerous sources. Netcool/OMNIbus is used to collect all events that are related to the business application landscape and TBSM uses these events to determine the status of the business applications.

System Automation for Multiplatforms automates start or stop dependencies in business application landscapes, provides common operating, automatic recovery in failure situations, and gets an aggregated availability status. System Automation for Multiplatforms and System Automation for z/OS make individual components of the business application highly available, for instance a critical database.

System Automation for Multiplatforms can be used to integrate with TBSM by enriching TBSM service views with data from System Automation events.System Automation for Multiplatforms delivers a TBSM service template containing preconfigured rules how to map states from System Automation to TBSM service instances.

Prerequisites

Before you begin, install and configure the following products and test your installation:

- Configure and enable event forwarding to OMNIbus for System Automation for Multiplatforms events. For more information, see “Configuring OMNIbus to process System Automation events” on page 97 and “Enabling event generation” on page 100.
- Tivoli Business Service Manager (TBSM) V4.2.1 or higher
- Update the Netcool OMNIbus ObjectServer schema for TBSM.
  - If you have an existing OMNIbus server, import the schema files `tbsm_db_update.sql` and `ClearServiceDeps.auto`.
  - If OMNIbus is installed with TBSM, the TBSM installer performs the required schema updates.

You can find TBSM specific product information in the TivoliBusiness Service Manager. For more information about the installation of the product, refer to Tivoli Business Service Manager Knowledge Center.

Configuring TBSM

To simplify the process for defining and configuring services in TBSM, service templates can be defined for services instances with common behavior. Rather than define each of the services and their dependencies individually, one template can be created for a type of service and then be assigned to applicable services.

Service instances represent actual services that are assigned a template. The template defines how a service responds to incoming data and the status of other services. Services of the same type should be assigned to a common template. This allows to use the same template rules to evaluate the status of multiple services.

When you assign a template to a service, you tag the service with the template. Templates eliminate the necessity of creating the same rules for a service type more than once.

Service template for TBSM

System Automation for Multiplatforms provides a TBSM service template that is used for System Automation resources, which are displayed in a TBSM service tree.

The service template is named `EEZ_SystemAutomationResource`. It provides

- An incoming status rule that is named `SACompoundState`, which uses state change events, which come from System Automation for Multiplatforms resources to determine the overall state of services.
- Text-based incoming status rules, which export the System Automation observed state and other System Automation specific states of a resource, so that they can be used in TBSM views. For more information how to use the text-based incoming status rules, see “Customizing TBSM views to add information from System Automation” on page 107.

The `EEZ_SystemAutomationResource` service template contains an incoming status rule that is named `SACompoundState`, which determines the overall state of a service. If the service template has been assigned to a specific service instance, resource state change events, which come from System Automation for Multiplatforms influence the overall state of the service. Events are associated with a service instance if the `AlertKey` in the event matches the `AlertKey` defined as identifier for the service instance.

TBSM has three available overall states: Bad, Marginal, and Good. The following mapping is defined in the `SACompoundState` rule to map resource state change events from System Automation to an overall TBSM state for a service instance:

Table 31. Mapping of System Automation resource state change events to TBSM states

Event Severity	**TBSM State**
5 (Critical)	Bad (Red)
4 (Major)	Bad (Red)
3 (Minor)	Marginal (Yellow)
1 (Indeterminate)	Good (Green)

Since there is a one-to-one mapping from a resource’s compound state to the event severity, the System Automation compound state directly determines the TBSM state. For more information about the mapping of compound state to event severity, see “Compound state to severity mapping” on page 96.

Defining a System Automation service template in TBSM

The EEZ_SystemAutomationResource template is required to use System Automation events in TBSM, import the EEZ_SystemAutomationResource template into TBSM as follows:

1. Copy the file EEZ_SystemAutomationResource.radsh from the /integration directory of the System Automation for Multiplatforms product CD to a temporary directory where the TBSM data server is installed.
2. Open a command prompt on the TBSM data server system. Change to the directory to which you have copied EEZ_SystemAutomationResource.radsh and issue the following command:
   - **UNIX:**

     ```
     cat EEZ_SystemAutomationResource.radsh  |
     $TBSM_HOME/bin/rad_radshell
     ```

   - **Windows:**

     ```
     type EEZ_SystemAutomationResource.radsh |
     %TBSM_HOME%\bin\rad_radshell
     ```

The service template provided by System Automation for Multiplatforms is now defined in TBSM.

Defining trigger in Netcool/OMNIbus

In the OMNIbus ObjectServer, a new state change event for a resource replaces the previous event (event deduplication).

By default, TBSM processes only a deduplicated event when the value of the **Severity** field changed. In these cases, TBSM processes the deduplicated events and updates the service status. A status change is possible for a resource, which updates status fields that are used in the text-based incoming status rules, which are contained in the EEZ_SystemAutomationResource service template. But the severity value does not change because the compound state of the resource does not change. Define a trigger in OMNIbus, to ensure that TBSM updates the services in these cases as well.

The sa_db_tbsm_update.sql file is used to define the trigger that is named update_tbsm_service_on_sa_events in OMNIbus. This trigger ensures that TBSM reprocesses events if one of the states that are used in the text-based incoming status rules changes, even if the severity value does not change. Whenever you want to use the text-based incoming status rules included in the EEZ_SystemAutomationResource service template, create this trigger definition.

Enter the following command on the OMNIbus server to define the trigger:

UNIX:

```
$OMNIHOME/bin/nco_sql -server NCOMS -username root < sa_db_tbsm_update.sql
```

Windows:

```
%NCHOME%\bin\redist\isql -S NCOMS -U root < sa_db_tbsm_update.sql
```

Enter userID and password when prompted.

sa_db_tbsm_update.sql is included with System Automation for Multiplatforms and can be found in the directory /integration on the product DVD.

Integrating System Automation resources and TBSM

If you want to add System Automation resources to a TBSM service tree, you have to manually create a service instance in TBSM and then assign the System Automation service template. This is described in “Assigning the service template to a service instance” on page 106. You also do this if you want to enrich service instances which already exist in a TBSM service tree with information from System Automation events.

Note: If you are also using System Automation Application Manager, you can make use of its Discovery Library Adapter in order to create service instances automatically for resources that are managed by System Automation Application Manager.

Assigning the service template to a service instance

A service template consists of rules that can be applied for service instances. A template can be used for more than one instance. If you want to assign the `EEZ_SystemAutomationResource` template to a service, you can tag the service with the template.

Proceed as follows:

1. Tag services using the `EEZ_SystemAutomationResource` template to make the defined incoming status rules available to these services.
   a. In the **Service Navigation** Portlet, select the **Service Name** for which you want to assign the System Automation specific service template `EEZ_SystemAutomationResource`.
   b. Select the **Edit Service** tab in the **Service Editor** to edit the service.
   c. Select the **Templates** tab. You can see the following two lists:
      - **Available Templates**: Displays all templates which you have the permission to assign to the selected service instance.
      - **Selected Templates**: Displays all templates assigned to the service.
   d. To assign the System Automation template to a service, select the `EEZ_SystemAutomationResource` template from the **Available Templates** list. Click the arrow button **>>** to move the template to the **Selected Templates** list.
2. Configure the **Identification Field** values for this service. TBSM uses the identification fields to map incoming events to a service instance.
   a. Select the **Edit Service** tab.
   b. Select the **Identification Fields** tab which provides the rules defined in the `EEZ_SystemAutomationResource` template and the identification field values required to map an event to the selected service instance. The rules contained in the `EEZ_SystemAutomationResource` template use the `AlertKey` event attribute as identifier. By default, the value for each identification field is the value entered in the **Service Name** field.
   c. Enter the correct `AlertKey` attribute value that corresponds to the selected service. The `AlertKey` must contain the unique System Automation resource key formatted as CDM `SourceToken`. The structure is defined like this:

```
EEZResourceKey,DN={DomainName},NN={NodeName},
RN={ResourceName},RC={ResourceClass}
```

   You may consider to open one of the events of the resource and copy and paste the `AlertKey` value from the event to avoid typing errors. Examples for valid `AlertKey` values:

   Resource
   Constituent or fixed resource. , which is displayed by `lssam` as `IBM.Application:db2-rs:saxb32c`.

   AlertKey:

```
EEZResourceKey,DN={DB2Cluster},NN={saxb32c},RN={db2- rs},
RC={IBM.Application}
```

   Move group
   Floating resource. The domain `DB2Cluster` is displayed by `lssam` as: `IBM.Application:db2-rs`

   AlertKey:

```
EEZResourceKey,DN={DB2Cluster},NN={},RN={db2- rs},
RC={IBM.Application}
```

Resource group

The domain is `DB2Cluster`, which is displayed by `lssam` as: `IBM.ResourceGroup:DB2`.

AlertKey:

```
EEZResourceKey,DN={DB2Cluster},NN={},RN={DB2},
RC={IBM.ResourceGroup}
```

d. Click **Save** to apply your changes.

Whenever new System Automation for Multiplatforms state change events are received for the service which match the specified `AlertKey`, TBSM will now process the incoming status rules and potentially change the overall state of the service based on the event severity.

Customizing TBSM views to add information from System Automation

The `EEZ_SystemAutomationResource` service template contains text-based incoming status rules which retrieve the System Automation Observed State and other System Automation specific states of a resource. This information can be used in TBSM Views in order to enrich service instances with information coming from System Automation for Multiplatforms.

The following text-based incoming status rules are available:

Table 32. Text-based incoming status rules for TBSM

Rule Name	Description
SAObservedStateValue	Retrieves the field `SAObservedState` from a resource status change event. Possible values: • Unknown • Online • Offline • Starting • Stopping • NotApplicable
SADesiredStateValue	Retrieves the field `SADesiredState` from a resource status change event. Possible values: • Online • Offline • NoChange (i.e. the resource's automation goal cannot be changed by an operator)
SAOperationalStateValue	Retrieve the field `SAOperationalStateValue` from a resource status change event. List of Operational State values giving more fine-grained information about the current state of the resource. For a list of possible values, see the `SystemAutomation.baroc` file.

Table 32. Text-based incoming status rules for TBSM (continued)	
Rule Name	**Description**
SACompoundStateValue	Retrieve the field `SACompoundStateValue` from a resource status change event. Compound State indicating whether the resource is working as desired or has encountered an error. Possible values: • Ok • Warning • Error • Fatal
SAExcludedFromAutomationValue	Retrieve the field `SAExcludedFromAutomationValue` from a resource status change event. Flag indicating if the resource is excluded from automation (i.e. automation is suspended). Possible values: • NotExcluded • Excluded

Adding columns for additional System Automation information to a TBSM service tree

You can modify the columns of custom trees displayed in TBSM in the

- **Service Navigation** Portlet
- **Service Tree** Portlet

The default **Service Navigation** Portlet has three columns:

- **State**
- **Time**
- **Events**

You can modify, delete, and add tree columns with the **Tree Template Editor** . The **Tree Template Editor** is available from the **Services** toolbar in the **Service Navigation** Portlet. You can add a new tree template to the **Service Navigation** Portlet. For each custom column, use the **Tree Template Editor** to specify the rule data you want to display in the column.

Adding columns:

This capability can be used to add columns for any of the provided text-based incoming status rules defined by the `EEZ_SystemAutomationResource` template. For example, you can define a column which displays the current Observed State coming from System Automation for each service instance that has the `EEZ_SystemAutomationResource` template assigned. Perform the following steps:

1. Click the **Tree Template Editor** button in the toolbar of the Service Navigation portlet.
2. Select the tree template you want to modify in the **Tree Template Name** drop-down list.
3. Click the **Add New Tree Column** button in the Column Configuration section.
4. Type the name you want to use in the blank field for the new column, for example “Availability State”.
5. Adjust the column position and width as appropriate
6. In the **Service Template Selection section**, select the `EEZ_SystemAutomationResource` template.
7. In the **Service Template Rule Mapping**, select the `EEZ_SystemAutomationResource` template in the Active Template list.

8. For each rule that you want to display in a service tree column, select the **Display** check box and choose a column from the drop-down box to display the output value. In this example, select the **Display** check box for the attribute `@SAObservedStateValue` and choose the **Availability State** column from the drop-down box of that row.
9. Click **OK** to save the changes to the tree template.

The following figure shows a screen capture of the tree template editor. A new column **Availability State** is added showing the System Automation Observed State:

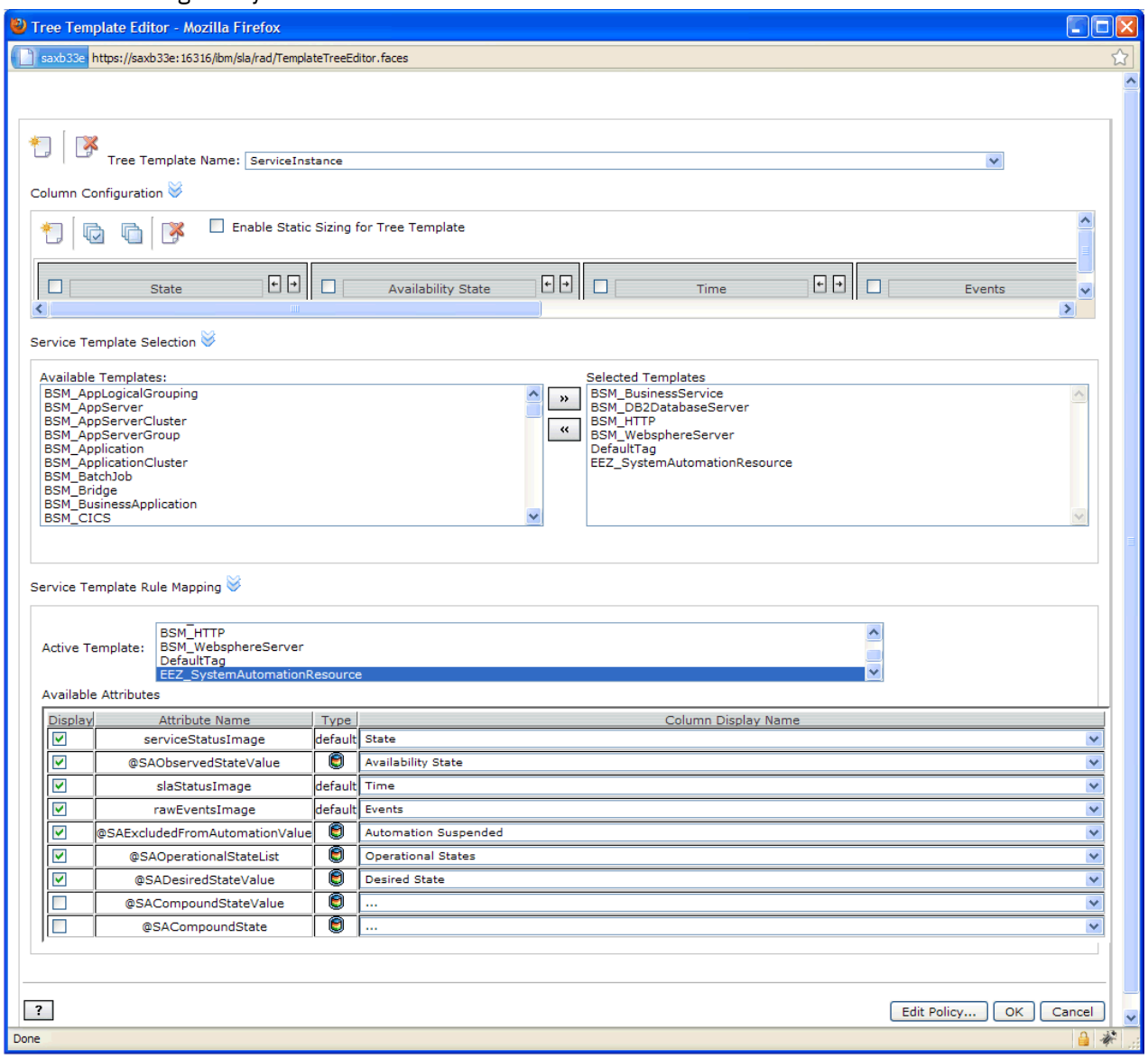

Figure 18. Tree template editor

To view the updated Services Tree, refresh the Service Navigation portlet. The new column now occurs showing the output of the incoming status rule that you have selected.

Note: You have to create new resource status change events in order to update the state information displayed in TBSM. Old events are not processed again.

Using the TBSM policy editor:

Optionally, you can format column values using the TBSM policy editor. For example, display the SA Observed State values in different colors. Proceed as follows:

1. Click the **Tree Template Editor** button in the toolbar of the Service Navigation portlet.
2. Choose the tree template you want to modify from the **Tree Template Name** drop-down list.

3. Click on the **Edit Policy...** button to open the policy that displays column values. The policy named `GetTreeColumnValue` is opened in the policy editor:

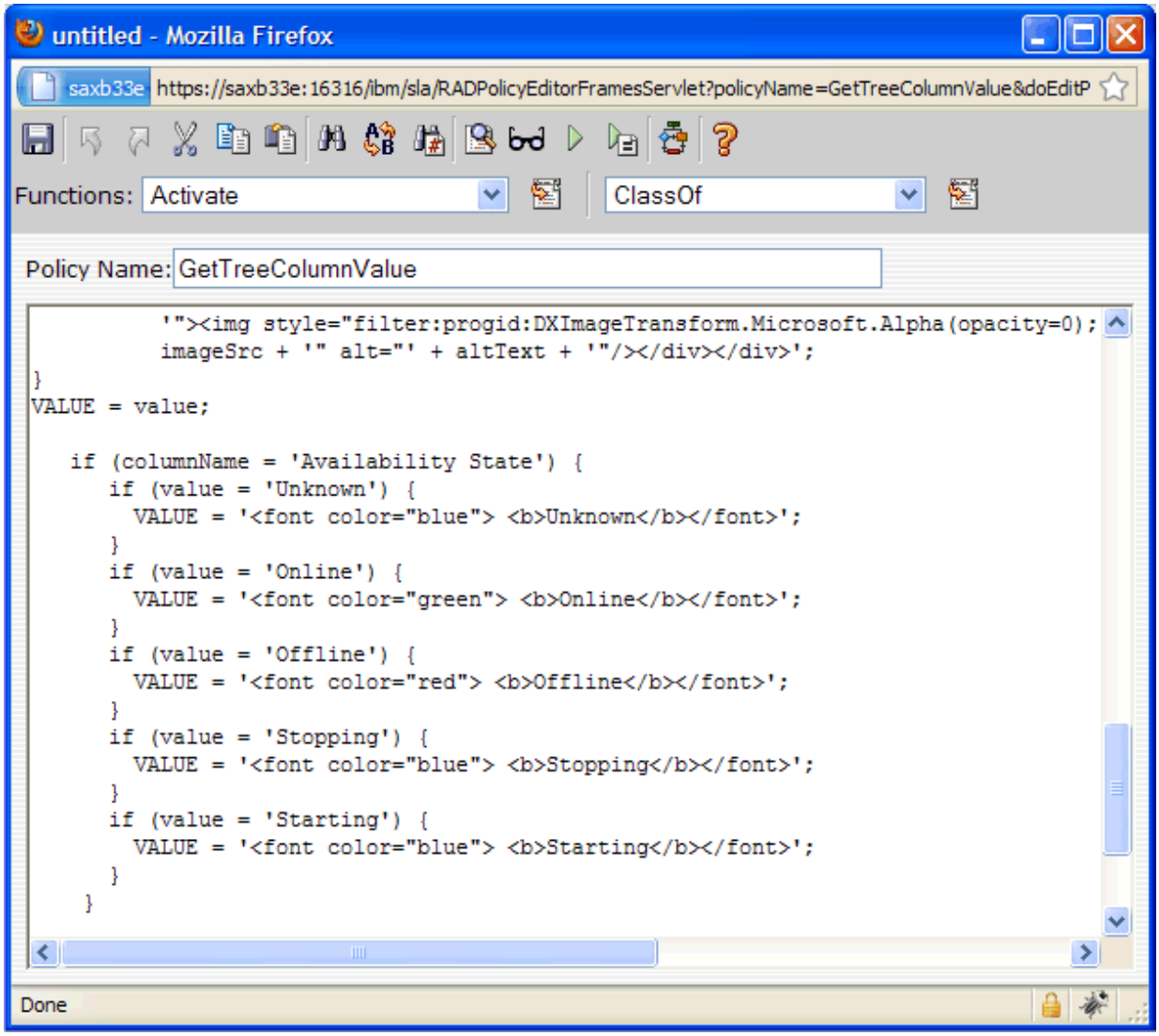

Figure 19. TBSM Tree Template Editor

4. Modify the policy. The following code snippet is as an example on how to change the color of the text-based output values. In this example, it is assumed that a column named “Availability State” has been defined showing the output of the SA`OservedState Rule`. Depending on the value of the observed state, the policy snippet returns the value in a different color:

```
if (columnName = 'Availability State') {
      if (value = 'Unknown') {
        VALUE = '<font color=“blue"> <b>Unknown</b></font>';
            }
            if (value = 'Online') {
        VALUE = '<font color=“green"> <b>Online</b></font>';
            }
            if (value = 'Offline') {
        VALUE = '<font color=“red"> <b>Offline</b></font>';
            }
            if (value = 'Stopping') {
        VALUE = '<font color=“blue"> <b>Stopping</b></font>';
            }
            if (value = 'Starting') {
        VALUE = '<font color=“blue"> <b>Starting</b></font>';
            }
```

5. Save the modified policy

Chapter 5. Securing

Securing your System Automation for Multiplatforms environment involves configuring Secure Socket Layer (SSL) connections and protecting your cluster environments from unauthorized access.

You can set up non-root security for the command line interface of System Automation for Multiplatforms on AIX, and Linux systems.

On Linux and AIX systems, by default only the user `root` has the required authority to complete operational tasks in System Automation for Multiplatforms and to change the automation policy of System Automation for Multiplatforms, while all other users have read access only.

Managing authorization for users accessing the cluster

The security concept of System Automation for Multiplatforms is based on the RSCT component RMC, which implements security authorization with an access control list (ACL) file. Specifically, RMC uses the ACL file on a particular node to determine the permissions that a user must have to access resource classes and their resource instances. Since the System Automation resource managers are internally implemented as an RMC application, the same set of ACL control rules must be used to allow non-root users to manage (define, undefine, or change) the System Automation-related resource classes (`IBM.ResourceGroup`, `IBM.ManagedRelationship`, `IBM.Equivalency`, `IBM.ManagedResource`, `IBM.CHARMControl`, `IBM.Application` and `IBM.ServiceIP`) and to start and stop the corresponding resource groups.

For detailed information about how to set up RMC ACL files, see the following sections in IBM RSCT Administration Guide:

- “Managing user access to resources using RMC ACL files” in Chapter 4 ("Managing and monitoring resources using RMC and resource managers")
- “Configuring the global and local authorization identity mappings” in Chapter 7 ("Understanding and administering cluster security services")

Setting up non-root user Ids for the command line interface

RSCT and RMC security authorization support manages user access based on individual resource classes and single nodes, for example, user access can be restricted to a specific RMC resource class on a particular node within the cluster. This level of authorization setting is complex and requires a clear understanding of the nature of each individual RMC resource class.

Therefore, you must create roles for a System Automation for Multiplatforms operator and a System Automation for Multiplatforms administrator with general settings that allow non-root users to manage all resource classes from any node that is defined within the cluster. Use the following procedure to create these two roles:

- `sa_admin` for an administrator
- `sa_operator` for an operator

The roles are described in more detail in the section: http://www.ibm.com/support/knowledgecenter/en/SSRM2X_4.1.0/com.ibm.samp.doc_4.1/sampugbug_limit_non-root.html

System Automation for Multiplatforms version 4.1.0.4 or higher provides the script samnonrootuser to perform this non-root user setup. The script requires an existing user, and then adjusts file permissions and ACL files to define the user as either 'sa_admin', or 'sa_operator'.

If the installed System Automation version is lower than 4.1.0.4, the manual setup as described below has to be performed :

To create the roles, perform the following steps (note that root authority is required). This example shows the commands that you must run in a Linux environment:

1. Create the user IDs that are authorized to manage System Automation for Multiplatforms on all nodes:

```
# /usr/sbin/useradd ernie
# /usr/sbin/useradd bert
```

2. Create a group for the user IDs on all nodes:

```
# /usr/sbin/groupadd sagroup
```

3. Add the group to the user IDs on all nodes:

```
# /usr/sbin/usermod –G sagroup ernie
# /usr/sbin/usermod –G sagroup bert
```

   Note: Make sure to set the following environment variable for all users of System Automation for Multiplatforms on all nodes (peer domain scope):

```
CT_MANAGEMENT_SCOPE=2
```

   You can set the variable permanently if you set it in the user profile.

4. Change the group ownership of the file `/var/ct/IBM.RecoveryRM.log`.

   The file is used to track the System Automation for Multiplatforms history. All commands that modify the resources of the automation manager (`IBM.RecoveryRM`) are logged to that file.

   By default, the file is owned by the user group `root`:

```
-rw-r--r--  1 root root   204 Oct 4 22:00 /var/ct/IBM.RecoveryRM.log
```

   Change the group ownership to `sagroup`:

```
/bin/chgrp sagroup /var/ct/IBM.RecoveryRM.log
```

   Change the file permission to 664:

```
# /bin/chmod 664 /var/ct/IBM.RecoveryRM.log
       -rw-rw-r-- 1 root sagroup 204 Oct 4 22:00 /var/ct/IBM.RecoveryRM.log
```

   Note: If the file `/var/ct/IBM.RecoveryRM.log` does not exist after the initial installation of System Automation for Multiplatforms, you can create a dummy file by running the `/usr/bin/touch` command:

```
# /usr/bin/touch /var/ct/IBM.RecoveryRM.log
```

5. Modify the file `/var/ct/cfg/ctsec_map.global` on all nodes.

   You must add the following entries for the user IDs `ernie` and `bert` to the RSCT global authorization identity mapping file (`/var/ct/cfg/ctsec_map.global`) on every node in the cluster. Add the new entries above the entry for the user `clusteruser`:

```
unix:ernie@<cluster>=sa_operator
unix:ernie@<any_cluster>=sa_operator
unix:bert@<cluster>=sa_admin
unix:bert@<any_cluster>=sa_admin
unix:bert@<iw>=sa_admin
..
unix:*@*=clusteruser
```

   The file is used to map a local user ID on a node to a global user ID within the System Automation for Multiplatforms domain. In the example, the local user ID `ernie` is mapped to the global user ID `sa_operator`, and the local user ID `bert` is mapped to the global user ID `sa_admin`.

   You can authorize more local user IDs for System Automation for Multiplatforms by adding lines to this global map file (on all nodes), and mapping them to the wanted role operator or administrator.

Note: If the file //var/ct/cfg/ctsec_map.global does not exist on a node, copy the default file /usr/sbin/rsct/cfg/ctsec_map.global to the directory /var/ct/cfg and add the new entries to the file /var/ct/cfg/ctsec_map.global. Do not remove any entries from the file /var/ct/cfg/ctsec_map.global that exist in the default file you copied. The /var/ct/cfg/ctsec_map.global files on all nodes within the cluster must be identical. Always add new IDs for non-root users above the entries for the user clusteruser.

6. Modify the file /var/ct/cfg/ctrmc.acls on all nodes. You must add the following entries for the global user IDs sa_operator and sa_admin to the RMC ACL file (/var/ct/cfg/ctrmc.acls) on every node in the cluster and remove the line that starts with LOCALHOST, for example:

```
The following stanza contains default ACL entries.
# These entries are appended
# to each ACL defined for a resource class and
# are examined after any entries
# explicitly defined for a resource class
# by the stanzas in this file,
# including the OTHER stanza.
DEFAULT
root@LOCALHOST * rw
none:root * rw // give root access to all
none:sa_admin * rw // append this row for saadmin
none:sa_operator * rso // append this row for saoperator
```

7. When you completed the required modifications, run the following command on every node in the cluster to activate the changes:

```
# /usr/bin/refresh -s ctrmc
```

8. Extra changes that are required to use **sampolicy** and ***samadapter** commands:

   a. Access to configuration files:

   ```
   # /bin/chgrp -R sagroup /opt/IBM/tsamp/sam/cfg
   # /bin/chmod g+ws /opt/IBM/tsamp/sam/cfg
   # /bin/chmod g+w /opt/IBM/tsamp/sam/cfg/*
   ```

   b. Access to log files:

   ```
   # /bin/chgrp -R sagroup /var/ibm/tivoli/common/eez/logs
   # /bin/chmod g+ws /var/ibm/tivoli/common/eez/logs
   # /bin/chmod g+w /var/ibm/tivoli/common/eez/logs/*
   ```

   c. Access to configuration files in the /etc directory. If there is no directory /etc/opt/IBM/tsamp/sam/cfg, create it by using

   ```
   # /bin/mkdir -p /etc/opt/IBM/tsamp/sam/cfg
   ```

   Then, set the appropriate permissions:

   ```
   # /bin/chgrp -R sagroup /etc/opt/IBM/tsamp/sam/cfg
   # /bin/chmod g+ws /etc/opt/IBM/tsamp/sam/cfg
   # /bin/chmod g+w /etc/opt/IBM/tsamp/sam/cfg/*
   ```

9. Optional adjustments that are required for working with the sam.policies package: Pre-canned policies for various applications are provided in the installation package sam.policies, which can be downloaded from IBM Integrated Service Management Library.
10. To allow a user who has the sa_admin role to set up these pre-canned policies, the permissions, and the ownership of the /usr/sbin/rsct/sapolicies directory must be changed after the sam.policies package is installed on all nodes:

```
# chmod –R 2775 /usr/sbin/rsct/sapolicies
# chgrp -R sagroup /usr/sbin/rsct/sapolicies
```

When you completed the steps successfully, the local users ernie and bert can run operational tasks of System Automation for Multiplatforms, such as issuing start and stop requests against resources, and

the local user `bert` can also run administrative tasks of System Automation for Multiplatforms, such as defining and modifying policies.

Modified default authorization for non-root users using RSCT Level 2.5.4.0 or higher

Starting with RSCT level 2.5.4.0 (AIX 6, and Linux) a change was introduced that prevents non-root users from running commands to list resources. The appropriate permissions are automatically configured if a new domain is created.

If you migrate an existing domain to this RSCT level, the appropriate permissions to run commands like `lssam` or `lsrg -m` are not automatically configured for non-root users. Depending on your RSCT level, choose the appropriate actions to adjust the configuration:

The RSCT level is equal to or higher than 2.5.5.2 (AIX 6, and Linux:
Create another domain that implicitly adjusts the configuration. Do not start the new domain. You can remove it later.

Alternatively, or if the RSCT level is lower than 2.4.13.2:
Use the following commands to adjust the configuration on all nodes as user root:

1. Edit the file `/usr/sbin/rsct/cfg/ctsec_map.global` and add the following content if it does not exist:

   ```
   unix:*@*=clusteruser
   ```

2. Create a file `/tmp/addacl` and add the following content:

   ```
   DEFAULT
      none:clusteruser * r
   ```

3. Adjust the `acl` file by running the following command:

   ```
   /usr/sbin/rsct/install/bin/chrmcacl -a < /tmp/addacl
   ```

4. Refresh the `ctrmc` sub system for the changes to become effective:

   ```
   refresh -s ctrmc
   ```

Non-root users are now able to use commands like `lssam` or `lsrg -m` as with earlier RSCT levels.

Limitations of the non root security setup

The following list summarizes the limitations of the non root security setup:

- A regular user cannot view the contents of the RMC resource manager trace file (for example, the trace of the`IBM.RecoveryRMd` daemon).

  All RMC Resource Manager daemons use the RMC framework library utility to create trace files and core images under the `/var/ct/<cluster>` directory. Since these resource managers can be started only by a superuser (user ID `root`) through the **/usr/bin/startsrc** command, the files that are created belong to the user ID `root`.

  All non root users cannot collect debug and trace information by using the **/usr/sbin/rsct/bin/ctsnap** command.

  To allow non root users to collect traces or `ctsnap` debug data or both, a mechanism like "sudo" must be implemented for these users and commands.
- The following commands can be started only with `root` authority because they use Tivoli logging, which works properly only if the log files are maintained with `root` rights:
  - The **sampolicy** command.

  - The **samadapter** command to start the end-to-end automation adapter.
  - The **samlicm** command to install or upgrade a license.
- The granularity of the ACL objects is based on resource classes, not on resources. This means that a regular user is either allowed to modify resources of a resource class or not, but it is not possible to grant or deny permissions on a resource basis, for example, a database administrator cannot be authorized only for database resources.
- The "sa_operator" role can modify resources by changing attribute values for the resources. This is a result of the "s" permission, which is needed for issuing System Automation for Multiplatforms requests. Without the "s" permission, users who have this role would not be able to perform any useful task. With the "s" permission they are allowed to set and change attributes.

The following table shows which role or authority is required to perform typical System Automation for Multiplatforms tasks.

Table 33. Authorizations and roles for performing System Automation for Multiplatforms tasks

Task	Authority	Roles	Permissions
Product and product license installation	root	System Administrator	Installing and upgrading System Automation for Multiplatforms and the product license.
Cluster management	root / sa_admin	System Administrator / System Automation for Multiplatforms Administrator	Defining, starting, stopping, and monitoring clusters and individual RMC Resource Managers
Resource definition and System Automation for Multiplatforms policy definition	root / sa_admin	System Administrator / System Automation for Multiplatforms Administrator	Defining, removing, changing resources, and setting up automation policies
Automation operation	root / sa_admin / sa_operator	System Administrator / System Automation for Multiplatforms Administrator and Operator	Issuing Online and Offline request, and resetting and monitoring resource groups and individual resources
Collecting trace and debug data for problem determination	root	System Administrator	Access to all system and application trace (log) files. (see the list of limitations)
Security setup	root	System Administrator	Defining, changing, and removing the security setup that is described in this section.
Adapter setup	root / sa_admin	System Administrator / System Automation for Multiplatforms Administrator	Defining, changing, and removing the configuration of the end-to-end automation

Securing the connection to the end-to-end automation adapter using SSL

Configure Secure Socket Layer (SSL) in your environment for communication between the System Automation Application Manager end-to-end automation server and the System Automation for Multiplatforms end-to-end automation adapter.

This topic describes how to secure the connection between the System Automation Application Manager server and the end-to-end automation adapter. The connection between the System Automation Application Manager server and the automation adapter is a two-way communication and all queries and actions are secured with SSL encryption. Sending EIF events from the automation adapter to the System Automation Application Manager server is not secured. For more information about securing this connection, see *IBM Tivoli System Automation Application Manager Installation and Configuration Guide*.

Generate Keystore and Truststore with SSL public and private keys

Generate the following files:

- **Truststore**: Contains the public keys for Application Manager and the FLA adapters.
- **Application Manager Keystore:** Contains the private key for Application Manager.
- **Adapter Keystore** : Generate one per adapter. Contains the private key for a FLA adapter.

Figure 20 on page 117 shows an overview of the involved components, files and steps to generate the files. In the following, the term *operations console* refers to the System Automation Application Manager operations console.

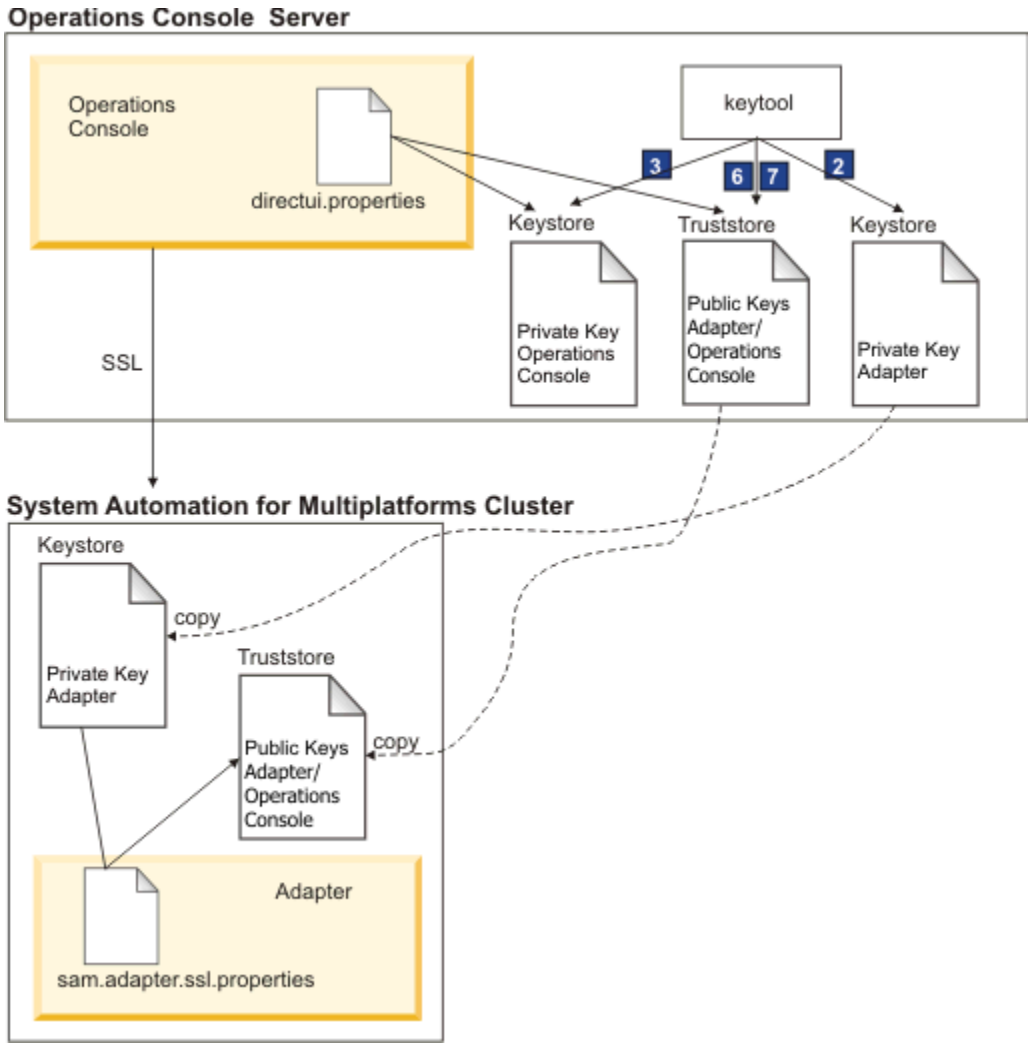

Figure 20. Keystore and Truststore generation using SSL

Generate the truststore and the keystore performing the following steps. The keys expire after 25 years with the default validity set to 9125. Make sure that the passphrase is at least 6 characters long. The numbers of the steps relate to the numbers in Figure 20 on page 117. The values that are used are sample or default values.

1. **Set variables:**

```
# java keytool from the operations console install directory
OC_INSTALL_DIR=/opt/IBM/tsamp/eez/jre/bin/keytool
# Operations console config file directory
OC_CONFIG_DIR=/opt/IBM/tsamp/eez/ewas/AppServer/profiles/AppSrv01/Tivoli/EEZ
# keys will expire in 25 years
KEY_VALIDITY_DAYS=9125
# passphrase at least 6 characters
PASSPHRASE=passphrase
```

2. **Generate keystore with public and private keys for the automation adapter:**

```
${JAVA_KEYTOOL} -genkey -keyalg RSA -validity ${KEY_VALIDITY_DAYS} \
          -alias samadapter -keypass ${PASSPHRASE} -storepass ${PASSPHRASE} \
          -dname "cn=SAAM Adapter, ou=Tivoli System Automation, o=IBM, c=US" \
          -keystore ${OC_CONFIG_DIR}/ssl/sam.ssl.adapter.keystore.jks
```

3. **Generate keystore with public and private keys for the operations console:**

```
{JAVA_KEYTOOL} -genkey -keyalg RSA -validity ${KEY_VALIDITY_DAYS} \
          -alias samoperationsconsole -keypass ${PASSPHRASE} -storepass ${PASSPHRASE} \
```

```
          -dname "cn=SAAM Server, ou=Tivoli System Automation, o=IBM, c=US" \
          -keystore "${OC_CONFIG_DIR}/ssl/sam.ssl.operationsconsole.keystore.jks"
```

4. **Export certificate file with public key for the automation adapter:**

```
${JAVA_KEYTOOL} -export -alias samadapter \
          -file ${OC_CONFIG_DIR}/ssl/samadapter.cer -storepass ${PASSPHRASE} \
          -keystore "${OC_CONFIG_DIR}/ssl/sam.ssl.adapter.keystore.jks"
```

5. **Export certificate file with public key for the operations console:**

```
${JAVA_KEYTOOL} -export -alias eezoperationsconsole \
          -file "${OC_CONFIG_DIR}/ssl/eezoperationsconsole.cer" -storepass ${PASSPHRASE} \
          -keystore "${OC_CONFIG_DIR}/ssl/sam.ssl.operationsconsole.keystore.jks"
```

6. **Generate authorized keys truststore and import certificate with public key for the automation adapter:**

```
${JAVA_KEYTOOL} -import -noprompt -alias samadapter \
          -file "${OC_CONFIG_DIR}/ssl/samadapter.cer" -storepass ${PASSPHRASE} \
          -keystore "${OC_CONFIG_DIR}/ssl/sam.ssl.authorizedkeys.truststore.jks"
```

7. **Generate authorized keys truststore and import certificate with public key for the operations console:**

```
{JAVA_KEYTOOL} -import -noprompt -alias samoperationsconsole \
          -file "${OC_CONFIG_DIR}/ssl/samoperationsconsole.cer" -storepass ${PASSPHRASE} \
          -keystore "${OC_CONFIG_DIR}/ssl/sam.ssl.authorizedkeys.truststore.jks"
```

8. **Delete certificate file for the automation adapter. The certificate file is not needed anymore at runtime:**

```
rm ${OC_CONFIG_DIR}/ssl/samadapter.cer
```

9. **Delete certificate file for the operations console. The certificate file is not needed anymore at runtime:**

```
rm ${OC_CONFIG_DIR}/ssl/samoperationsconsole.cer
```

Enable SSL security in automation adapter configurations

Perform the following steps to enable SSL security in the automation adapter configurations.

1. **Copy the authorized keys truststore file to all nodes in the IBM Tivoli System Automation for Multiplatforms cluster:**

```
scp ${OC_CONFIG_DIR}/ssl/sam.ssl.authorizedkeys.truststore.jks \
    root@<adapter-nodename>:/etc/opt/IBM/tsamp/eez/cfg/ssl/sam.ssl.authorizedkeys.truststore.jks
```

2. **Copy the adapter keystore file to all nodes in the IBM Tivoli System Automation for Multiplatforms cluster:**

```
cp ${OC_CONFIG_DIR}/ssl/sam.ssl.adapter.keystore.jks \
    root@<adapter-nodename>:/etc/opt/IBM/tsamp/sam/cfg/ssl/sam.ssl.adapter.keystore.jks
```

3. **Start the configuration utility.**

   Enter the command `cfgsamadapter`.

4. **Specify the parameters:**

   On the main window of the configuration dialog, click **Configure**. Specify the following parameters on the **Security** tab described in “Security tab” on page 71. Values below are sample values.

   - Truststore: `/etc/opt/IBM/tsamp/sam/cfg/ssl/sam.ssl.authorizedkeys.truststore.jks`
   - Keystore: `/etc/opt/IBM/tsamp/sam/cfg/ssl/sam.ssl.adapter.keystore.jks`

- Keystore password: `passphrase`
- Certificate alias: `samadapter`

Click **Save** to store the configuration changes.

5. On the main window of the configuration dialog, click **Replicate**. Replicate the configuration files to the other nodes in the cluster of IBM Tivoli System Automation for Multiplatforms cluster including the SSL configuration.
6. Restart the automation adapter using the `samadapter` command that is used to control the automation adapter. This activates the SSL configuration.
7. Restart the System Automation Application Manager server to activate the SSL configuration.

   Use the following commands to start or stop the System Automation Application Manager server manually:

   Start

   ```
   /opt/IBM/WebSphere/AppServer/bin/startServer.sh server1
   ```

   Stop

   ```
   /opt/IBM/WebSphere/AppServer/bin/stopServer.sh server1
   ```

   Note: The WebSphere Application Server administrative user ID and password are required to stop the System Automation Application Manager server.

Using IBM Support Assistant

IBM Support Assistant is a free, standalone application that you can install on any workstation. IBM Support Assistant saves you time searching product, support, and educational resources and helps you gather support information when you need to open a problem management record (PMR) or Electronic Tracking Record (ETR), which you can then use to track the problem.

You can then enhance the application by installing product-specific plug-in modules for the IBM products you use. The product-specific plug-in for Tivoli System Automation for Multiplatforms provides you with the following resources:

- Support links
- Education links
- Ability to submit problem management reports
- Capability to collect traces

Installing IBM Support Assistant and the Tivoli System Automation for Multiplatforms plug-in

To install the IBM Support Assistant V4.1, complete these steps:

- Go to the IBM Support Assistant Web Site:

  ```
  www.ibm.com/software/support/isa/
  ```

- Download the installation package for your platform. Note that you will need to sign in with an IBM user ID and password (for example, a MySupport or developerWorks® user ID). If you do not already have an IBM user ID, you may complete the free registration process to obtain one.
- Uncompress the installation package to a temporary directory.
- Follow the instructions in the *Installation and Troubleshooting Guide*, included in the installation package, to install the IBM Support Assistant.

To install the plug-in for Tivoli System Automation for Multiplatforms, complete these steps:

1. Start the IBM Support Assistant application. IBM Support Assistant is a Web application that is displayed in the default, system configured Web-browser.
2. Click the **Updater** tab within IBM Support Assistant.
3. Click the **New Products and Tools** tab. The plug-in modules are listed by product family.
4. Select **Tivoli > Tivoli Tivoli System Automation for Multiplatforms**.
5. Select the features you want to install and click **Install**. Be sure to read the license information and the usage instructions.
6. Restart IBM Support Assistant.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie New York 12601-5400
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

- IBM, the IBM logo, ibm.com, AIX, DB2, developerWorks, HACMP, NetView, Tivoli, Tivoli Enterprise, Tivoli Enterprise Console, WebSphere, and z/OS are trademarks of International Business Machines Corporation in the United States, other countries, or both. IBM Redbooks and the IBM Redbooks logo are registered trademarks of IBM.
- Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.
- Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Red Hat and all Red Hat-based trademarks are trademarks or registered trademarks of Red Hat, Inc., in the United States and other countries.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Other company, product, and service names may be trademarks or service marks of others.

Index

K

L

M

N

O

P

R

S

T

U

V

W

X

Z

Product Number: 5724-M00

SC34-2699-04